MICROCOMPUTADOR CURSO PRÁTICO

TIRE O MÁXIMO DO MICRO EM 30 SEMANAS

PUBLICAÇÃO SEMANAL ILUSTRADA

Cr$ 10.250

21

O som eletrônico • Atenção aos robôs
Modelos da vida real

rioGráfica

MANAUS, ALTAMIRA, MACAPÁ, JI.PARANÁ, PORTO VELHO, RIO BRANCO, SANTARÉM, BOA VISTA, SINOP, ALTA FLORESTA: Cr$ 13.330

MICROCOMPUTADOR CURSO PRÁTICO

APLICAÇÕES

Música em software 417

Robôs na trilha 422

SOFTWARE

Idéias em ordem 420

RAIO X

Wren Executive 425

PROJETOS DE PROGRAMAS

Aventuras e simulações 428

CHIPS & BYTES

Redes locais 430

Escher eletrônico 435

LINGUAGENS

Byte a byte 433

PERFIL

Microtec 436

TESTE

Depois de ler com atenção todo o fascículo, responda a estas perguntas. Veja a solução no próximo número, junto com um novo teste.

1) O Lunokhod 1, enviado pelos soviéticos à Lua em 1970, é um exemplo de que tipo de robô?
2) Qual o nome da notação para música eletrônica que oferece uma alternativa vantajosa ao pentagrama convencional?
3) Qual o recurso de hardware mais notável do micro Wren Executive?
4) Quais os três tipos mais comuns de arquitetura de rede local?

RESPOSTAS DO TESTE ANTERIOR

1) O microprocessador incorporado a algumas câmaras fotográficas modernas combina da melhor maneira abertura e velocidade de exposição.
2) O tablete gráfico Touchmaster tem área de trabalho maior que a do Koala, e é dotado de saídas serial e paralela, adequadas à maioria dos micros.
3) Um nybble é meio byte, ou seja, quatro bits tomados como uma unidade.
4) O SGBD citado, que relaciona dados entre vários arquivos, é o dBase II.

PLANO DA OBRA

MICROCOMPUTADOR - CURSO PRÁTICO é uma coleção de 30 fascículos de periodicidade semanal. Os 29 primeiros terão, cada um, vinte páginas internas (miolo) e quatro capas, além de uma introdução de dezesseis páginas acompanhando o fascículo 1. O fascículo 30 incluirá o índice geral da obra. Os miolos, encadernados, constituem dois volumes de um curso prático de microcomputação, com seções indicadas por tarjas de diferentes cores.
As duas primeiras capas são descartáveis; as 3.as e 4.as trazem programas de jogos. Encadernadas, formam um volume de cem páginas.

COMO ENCADERNAR

As capas duras, incluindo as guardas, estarão à venda simultaneamente com os fascículos 10 (**Volume 1**) e 30 (**Volume 2** e **Jogos**).
Volume 1 - Deve ser encadernado com os elementos dispostos nesta ordem: guardas; frontispício e introdução (publicados com o fascículo 1); miolos dos fascículos de 1 a 15; e guardas.
Volume 2 - Guardas; frontispício (que virá com o fascículo 30); miolos dos fascículos de 16 a 30; e guardas.
Jogos - Guardas; frontispício e sumário (que constituem as 3.as e 4.as capas do fascículo 30); 3.as e 4.as capas dos fascículos de 1 a 29; e guardas.

rioGráfica **CONSELHO DE ADMINISTRAÇÃO:** Roberto Irineu Marinho; João Roberto Marinho; José Roberto Marinho; Oscar Neves; **Diretoria:** Oscar Neves; Danilo Esteves Costa; João J. Noro; Marcos March; Nilo Sérgio de Almeida; **GRUPO EDUCAÇÃO E CULTURA - Divisão Comercial - Diretor:** Jaime Rodrigues; **Gerente de Planejamento:** João de Azeredo Coutinho; **Gerente de Vendas:** Rubens Barbosa; **Coordenadores de Promoção:** Paulo Cesar M. Seixas; Edgard Mello M. Neto; **Gerente de Circulação:** Norberto Martin; **Supervisor de Operações:** Abel Pereira Pinto; **Editorial - Diretor:** João J. Noro; **Editor-chefe:** Mauricio Rittner; **Administração Editorial:** Heitor de Souza Paixão; **Editor:** Jesse Navarro; **Editor-assistente:** Claudio Marcondes; **Revisores:** Isis Augusta Loyolla; Cacilda Guerra; Luiz Vicente Vieira Filho; Marcos Domingos Agathão; Maria da Graça Mendonça Couto; **Diretor de Arte:** Adalberto Cornavaca; **Chefe de Arte:** Bonifácio Duardes Miranda; **Chefe de Estúdio:** José Yuji Kuribayashi; **Assistentes de Arte:** Miguel Luis Escámez Simón; Edemilson Ribeiro da Silva; **Colaboradores - Editores-assistentes:** Isa Mara Lando; Luiz Carlos Pizarro Marin; **Texto:** Carlos Eduardo Silveira Matos; Jussara Maturo; Paulo Brito; **Preparação de Texto:** Luiz Carlos Cardoso; **Tradução:** Luiz Roberto de Godoy Vidal; Maria Clara Cescato; Maria Ester Menezes Martinho; Fernando Dorfman Knijnik; Newton Roberval Eichemberg; **Pesquisa:** Stella Maria Quentel; **Assessoria Técnica:** Geraldo Coen; Gustavo José Ferreira Grubba; José Rubens Salles Toledo; Lauro de Lauro; Lucas Kenishi Shirahata; Pierluigi Piazzi. As fotos não creditadas pertencem à obra original. Segue-se o crédito das demais: Cobra. © APSIF Copenhagen, 1984; © Orbis Publishing Ltd., 1984; © Editora Rio Gráfica Ltda., 1985, para a língua portuguesa. **Composição:** Editora Rio Gráfica Ltda. e AM Produções Gráficas Ltda. **Impressão:** JBIG.

NÚMEROS ATRASADOS - Você poderá comprar os exemplares que faltam em sua coleção, pelo preço do último fascículo posto à venda, em sua banca de jornal preferida. Caso não consiga obtê-los, faça sua solicitação por carta endereçada diretamente à Editora Rio Gráfica Ltda., rua Itapiru, 1209, Rio Comprido, Rio de Janeiro, CEP 20251. O atendimento será feito por via postal. Nas cidades do Rio de Janeiro e de São Paulo, as compras poderão ser efetuadas pessoalmente nos seguintes endereços: Rio de Janeiro - Rua Itapiru, 1209, Rio Comprido; São Paulo - Rua Frei Caneca, 1152, Consolação.

Distribuidor exclusivo para todo o Brasil: Fernando Chinaglia - Rua Teodoro da Silva, 907, Rio de Janeiro. Distribuidor para Portugal: Electroliber Ltda. - Rua Prof. Reinaldo dos Santos, 1488, Lisboa.

MÚSICA EM SOFTWARE

A MIDI, entre todos os sistemas disponíveis, é o que oferece aos músicos a mais ampla gama de aplicações, permitindo melhorar o desempenho tanto em execuções ao vivo como em estúdio.

A MIDI (Musical Instrument Digital Interface, "interface digital para instrumentos musicais") oferece aos estudantes vantagens inquestionáveis. Aprender a ler música costuma ser difícil, mesmo quando o iniciante já é músico competente. Os primeiros estágios, quando se fazem exercícios com sustenidos, bemóis e sinais de marcação de tempo, podem ser demorados e trabalhosos, e a correspondência entre os sinais no papel e o som que representam nem sempre é óbvia.

Grande parte do problema se deve à própria natureza da música: a harmonia depende da duração das notas e dos intervalos. Assim, se o iniciante precisa interromper a melodia para acompanhar a notação visual, todos os indicadores de intervalos de tempo na partitura perdem o sentido. De modo semelhante, ele pode levar algum tempo tentando interpretar determinada seqüência ou compasso; enquanto isso está sendo feito, a música terá avançado.

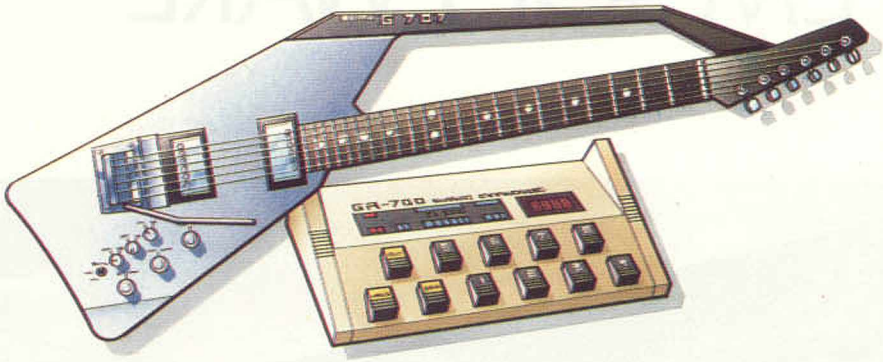
A MIDI elimina esses obstáculos, permitindo que uma execução em sintetizador seja armazenada na memória do computador. Com o software apropriado, obtém-se na tela do micro a representação gráfica da melodia que está sendo tocada. Trata-se de um auxílio inestimável para qualquer estudante de música: se um dó central (do piano) é tocado no sintetizador, a tela mostra um pentagrama com essa nota; tocando um acorde em si menor, seus componentes harmônicos — si, ré e fá — aparecem no monitor juntamente com a duração.

Essa idéia pode ser ampliada por um software que toca uma peça pré-programada no sintetizador interfaceado, enquanto a correspondente partitura completa vai aparecendo na tela. Nessa situação é possível interromper simultaneamente tanto a música quanto sua notação e reiniciá-las a partir de determinado compasso. Além disso, pode-se variar o som da música, como um todo, mudando os parâmetros de controle no sintetizador — introduzindo, assim, o usuário na arte do arranjo.

Atingindo certo grau de confiança na leitura da música, ficará mais fácil escrevê-la com auxílio do teclado alfanumérico de um computador. Isso pode envolver a introdução de dados de execução sem qualquer referência imediata a um som do sintetizador e a confrontação do resultado com aquilo que se pretendia. Uma vez adquirida essa habilidade, pode-se substituir a notação no pentagrama convencional em favor de outro sistema, como a MCL (Music Composition Language, "linguagem de composição musical").

Para a música eletrônica, a MCL revela-se o meio mais apropriado de introdução de dados,

Novos sons, novo estilo
Entre os sintetizadores da série DX da Yamaha, o DX7 incorpora um método novo de fabricar o som, e de síntese da FM, antes restrito a aparelhos de custo muito elevado. Em vez de partir de um som já existente e modificá-lo depois, passando-o por filtros ou ajustando controles de envoltória, o DX7 cria seus próprios sons complexos combinando de várias maneiras seus "operators" (memórias geradoras senoidais). Como resultado, o som obtido com o DX7 aproxima-se daquele dos instrumentos acústicos (vibrafone, cravo, baixo) de maneira bastante convincente. O DX7 também incorpora controle de respiração, de forma que um músico pode soprar num receptor e acrescentar aos sons de um saxofone ou de um trompete, por exemplo, variações que têm a respiração como fonte. O DX7 pode utilizar programas em ROM com características de som gravado, ou pode armazenar, em programas em RAM, sons criados pelo músico. O DX7 é vendido por um preço em torno de 2.000 dólares.



Som sintetizado

Existem vários sintetizadores ativados a partir das cordas de uma guitarra, em vez de um piano. O Roland GR700 por exemplo. Informações sonoras complexas são recebidas por um captador hexafônico (seis sons) e enviadas ao sintetizador, onde se acrescentam parâmetros que não provêm da guitarra. O SynthAxe, sintetizador controlado por um microprocessador 6809, emprega a mesma técnica, captando dados sonoros de um contato elétrico entre a corda e o traste. Sensores na parte superior da cravelha apanham variações de bending e de deslizamento. Um segundo conjunto de cordas e um pequeno teclado processam ainda mais o som.

uma vez que inclui a especificação de características exclusivas da produção eletrônica do som. Nenhum sistema padronizado foi desenvolvido ainda para a aplicação da MCL — as máquinas têm suas próprias MCLs. A notação em pauta, embora seja um guia visual imediato muito útil, é um sistema que antecede em muitos séculos a eletrônica, e não pode incluir todas as novas notações requeridas pela música eletrônica.

Para muitos donos de micros, compreender a notação em pauta ou a MCL será a chave para obter o máximo da MIDI. Nas execuções ao vivo, essa interface é, antes de tudo, um meio de integrar vários sintetizadores, seqüenciadores, computadores rítmicos e baterias eletrônicas, tudo gerenciado por um computador na forma de um sistema. A maior preocupação dos músicos que fazem uso extensivo do seqüenciamento é a possibilidade de perderem a sincronização entre as diversas unidades do sistema, com um resultante "colapso musical". Executantes famosos, como os Thompson Twins e Howard Jones, são conhecidos por rodarem fitas com trilhas de fundo gravadas em estúdio enquanto fazem execuções "ao vivo", evitando assim o risco.

Pelo menos em teoria, grupos que usam multi-sintetizadores deveriam estar mais livres de problemas graças à MIDI. Uma característica desse avanço é que tais grupos não darão mais a impressão de serem do tipo "multi-sintetizador". No início da década de 70, os sintetizadores eram, em geral, instrumentos de teclado com tantos botões e potenciômetros deslizantes que mais pareciam um painel de avião. Hoje, se um sintetizador de teclado está usando a MIDI para controlar um segundo ou um terceiro sintetizador, não há necessidade de mais teclados além daquele já existente no instrumento mestre.

À medida que o uso da MIDI se generaliza, mais "módulos sintetizadores" aparecem no mercado. São simplesmente as unidades geradoras e seqüenciadoras de som que antes faziam parte dos instrumentos de teclado. Como esses módulos têm pouco ou nenhum interesse visual, não é necessário que estejam presentes no palco.

Há outro recurso anterior à MIDI, mas que deve ganhar mais atenção à medida que o número de teclados diminui. Trata-se da possibilidade de se sintetizar o som eletrônico por dados provenientes de instrumentos de cordas e também por dados de controle, da respiração e da boca, provindos dos instrumentos de sopro. Em comparação ao ato mecânico de simplesmente pressionar uma nota no teclado, tanger uma corda ou fazer vibrar uma palheta envolve a transmissão de informações acústicas mais complexas ao instrumento em questão. Se esses dados são codificados digitalmente e transmitidos via MIDI a um módulo sintetizador fora do palco, não há razão para que um saxofonista não controle a eletrônica de todo o seu grupo — até mesmo a da bateria. A Yamaha incorporou controle de respiração em seu sintetizador DX7, e a SynthAxe — uma guitarra digital desenvolvida há pouco tempo — foi projetada de modo a usar a MIDI para controlar a saída de um Fairlight.

Isso tudo significa que, acionando a tecla do DX7 e controlando os dados da respiração, podem-se produzir as características únicas de execução de um saxofone, e, de modo semelhante, articula-se um som de "trombone" Fairlight dedilhando-se uma guitarra. Embora nenhum desses recursos seja iminente — afinal, a SynthAxe é uma "guitarra" muito cara —, eles indicam as tendências prováveis para execuções ao vivo num futuro próximo. Talvez ocorra uma redução gradual do número de teclados no palco, ganhando importância os instrumentos de cordas, sopro e, possivelmente, de percussão afinados, tais como os vibrafones; e, à medida que baratear a tecnologia da amostragem sonora, o som acústico poderá tornar-se predominante.

Nesse rumo, o final da década de 80 trará de volta, para alívio dos tradicionalistas, conjuntos de estilo jazzístico compostos de guitarra, saxofone, baixo elétrico e bateria. Mas eles poderão ficar confusos ao observar o guitarrista tocando um vibrafone invisível ou o saxofonista emitindo sons de bateria com seu sopro.

Para muitos grupos acostumados a execuções ao vivo, a experiência inicial com um avançado estúdio de gravação pode ser intimidante. Eles vão tomar contato com sistemas operacionais que nunca encontraram antes, e um produtor que pode não conhecer muito bem o trabalho ou as intenções dos músicos. A gravadora, porém, espera que, nesse ambiente estranho, eles produzam versões "maiores e melhores" de seus sucessos no palco. Isso equivale a colocar um grupo de teatro semiprofissional num filme de grande orçamento e esperar um sucesso de bilheteria instantâneo. Às vezes a transição é bem-sucedida, e todos os envolvidos ficam satisfeitos. Mas, com freqüência, as idéias originais se perdem num labirinto de aparelhos de estúdio, e o grupo produz um dispendioso fracasso.

É bastante comum ocorrer ruptura quando o grupo abandona seu próprio equipamento — e, com efeito, "seu" som — para adotar os instrumentos do estúdio. Uma idéia que funcionou num sintetizador Mini-Moog pode não funcionar num computador de amostragem digital Fairlight. E, se esse tipo de fracasso musical ocorre com muita freqüência, a justificativa para o uso do estúdio começa a enfraquecer.

No entanto, se os músicos do grupo já estiverem familiarizados com a MIDI e empregarem um microcomputador para armazenar seqüências e outros dados de controle musical, os procedimentos nos estúdios avançados não serão desconhecidos. A nível mais imediato, seria possível experimentar idéias utilizando uma sucessão de diferentes instrumentos de estúdio com um mínimo de dificuldade, e com o conhecimento prévio de que as idéias e as seqüências podem ser transformadas pela simples troca dos sintetizadores.

Ter a MIDI como técnica de apoio mostra-se algo inestimável também quando se enfrentam outros sistemas de estúdio diferentes daqueles diretamente envolvidos com a geração de som. Uma mesa de mixagem que utiliza a lógica do estado sólido, por exemplo, possui um computador para uso exclusivo, que chamará e rodará novamente qualquer série de decisões tomadas nos estágios finais da gravação — o que é conhecido como mixagem. Quando toda a música tiver sido gravada em 24 trilhas separadas — guitarra em uma, vocais de fundo em outra, vocais principais em mais três, e assim por diante —, começa a tarefa crucial de balancear e mixar todos os elementos.

Em geral, é nesse ponto que cada elemento musical recebe tratamento com qualquer "efeito" desejado, para que sobressaiam ou se misturem. Pode-se acrescentar reverberação a uma nota de trompete, apenas num determinado ponto; com 23 outras coisas acontecendo ao mesmo tempo, é fácil esquecer esse detalhe. Utilizar, enquanto se está mixando, um computador para manipular tais incidentes se parece com um seqüenciamento de MIDI em grande escala.

Outra técnica — originalmente desenvolvida para edição e sincronização de vídeo — que vem sendo empregada na produção musical consiste no uso do código de tempo. Trata-se de um recurso semelhante a um relógio digital com um sinal de disparador, mas gravado em fita. Utiliza palavras de 80 bits para fornecer dados de sincronismo quando se grava uma trilha sonora sobre seqüências de vídeo, e torna possível seqüenciar conjuntamente eventos musicais e imagens na tela.

Como se vê, os músicos têm boas razões práticas para adquirir instrumentos compatíveis com a MIDI. Mais que isso, porém, é a MIDI constituir uma boa introdução aos avançados sistemas musicais atualmente em uso.

Arte eletrônica
Laurie Anderson, poetisa e artista de vanguarda de Nova Iorque, combina uma mistura inusitada de sons e equipamento sonoro com filmes, fitas magnéticas e vídeo, criando um estilo único. Em algumas de suas canções, ela utiliza diretamente — ou como apoio de fundo — sons que vão desde sinos africanos aos instrumentos eletrônicos em "estado-de-arte", como o EMS Vocoder e o Synclavier, da New England Digital.

IDÉIAS EM ORDEM

Combinando recursos de processamento de texto e manipulação de listas, o programa BrainStorm — aqui analisado — proporciona flexibilidade na organização de idéias e de informações.

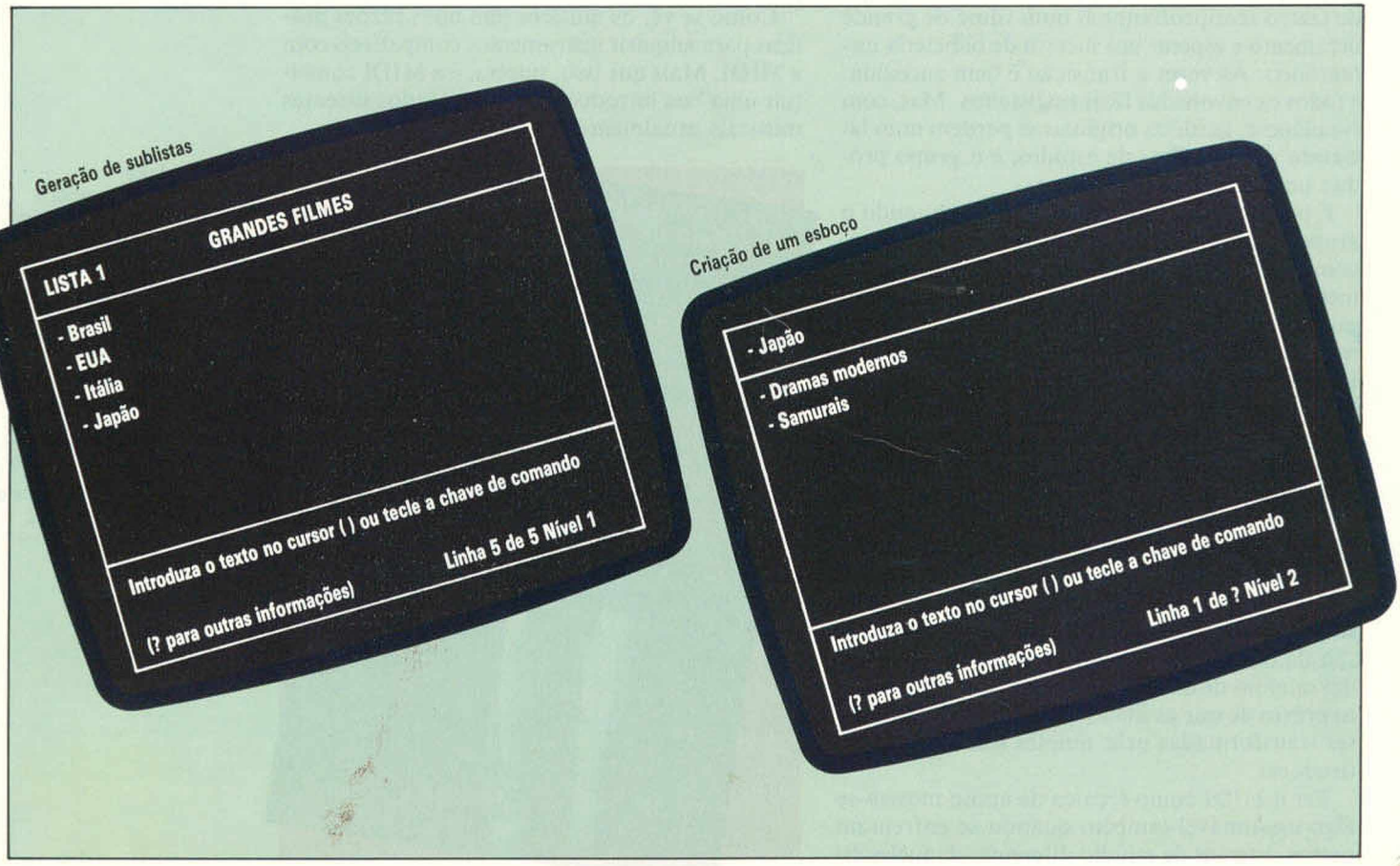
O BrainStorm, da firma inglesa Caxton Software, já foi cognominado de "processador de pensamentos", mas isso não significa que esse programa tente suplantar o cérebro humano. Na verdade, o BrainStorm ajuda a organizar os pensamentos. Para entender seu funcionamento, convém estabelecer uma analogia direta com os métodos pré-eletrônicos.

No desenvolvimento de qualquer projeto complexo, é comum a elaboração de listas de tarefas. Uma lista dos objetivos principais gera sublistas de como executar cada item arrolado, e assim por diante; o planejador pode ficar confuso com tantas folhas de papel. Algumas alterações na estrutura implicam tanto apaga-e-reescreve que o sistema se torna incontrolável. O BrainStorm simplifica tais alterações e acréscimos, da mesma forma que um processador de textos faculta eliminações e composições.

Qualquer item de uma lista ou sublista pode ser promovido a cabeçalho de uma sublista inferior; e, quando é necessário agrupar atividades similares de listas diferentes, elas poderão receber o mesmo nome e ser chamadas homônimas. Nesse caso, quaisquer introduções realizadas numa sublista encabeçada por um homônimo são agregadas a todas as outras listas que o usam.

Quem estiver familiarizado com os comandos de edição de um processador de textos, como o WordStar, achará muito fácil acostumar-se aos comandos de controle de cursor do BrainStorm. Digitando-se [CTRL] em conjunto com a letra [S], o cursor move-se para a esquerda. [CTRL]-[D], [CTRL]-[E] e [CTRL]-[X] fazem o cursor mover-se para a direita, para cima e para baixo, nessa ordem. Esses quatro comandos formam um padrão lógico no teclado, mas é possível redefini-los.

Mais difícil é acostumar-se com a possibilidade de elevar e baixar o cursor quando se introduz um texto, embora seja preciso alterar o modo — usando [CTRL]-[A] —, a fim de movê-lo para a esquerda ou para a direita numa linha.



Funcionamento do programa

O menu de abertura do BrainStorm oferece onze opções, escolhidas pela letra inicial do comando correspondente: Use, Load, Print, ID Drive (para utilizar outra unidade de disco), Save, Write (para gravar em disco), Directory, Xit (saída), Merge e Kill.

Inicialmente, o usuário digita [U] e passa a introduzir suas idéias como uma lista mais ou menos aleatória, chamada modelo. Por exemplo, um determinado modelo poderia consistir em:

```
Ler o manual
Digitar a lista
Digitar a sublista
Editar a lista
```

Os comandos de controle não aparecem imediatamente, mas são listados teclando-se [?]. Levando o cursor a qualquer item da lista e digitando [CTRL]-[R], promovemos esse item a cabeçalho de uma sublista. Da mesma forma, qualquer item dessa sublista pode transformar-se em cabeçalho de uma nova sublista, e assim por diante, até esgotar-se a memória. Digita-se [CTRL]-[C] para retorno à lista anterior.

Os itens tanto podem ser movimentados no interior de uma lista como transferidos para listas em outros níveis, quando rotulados com o caractere @. Para executar o movimento, digita-se em seguida [CTRL]-[G] (de Get) ou [CTRL]-[P] (de Put). Após a utilização de [CTRL]-[G], o caractere @ se move automaticamente para o item seguinte da lista; assim, um novo comando [CTRL]-[G] acessará o item seguinte, e assim por diante. Esse recurso permite transportar uma série de itens para sublista inferior ou superior.

Para inserir novos itens numa lista já existente, basta mover o cursor para o início da linha que deverá suceder o novo item e digitá-lo. Ao teclar [Return], o restante da lista descerá uma linha, dando espaço para o novo elemento. Com o comando [CTRL]-[A] altera-se qualquer item.

Itens homônimos de diferentes listas — por exemplo, datas — têm referência cruzada. Assim, se a lista exige a ocorrência de um evento em determinado dia — por exemplo, 1.º de janeiro —, podemos criar uma lista de eventos para tal data, que será automaticamente atualizada por meio da referência cruzada.

Pode-se criar qualquer número de homônimos e então acessá-los em seqüência, usando [CTRL]-[S] para a ocorrência seguinte e [CTRL]-[D] para a anterior. Após a última ocorrência, [CTRL]-[S] volta em loop para a primeira.

Impressão de listas

É possível imprimir as listas, normalmente formatadas com recuos para indicar seus níveis. Por exemplo:

```
Ler o manual
  Tirar o manual da caixa
    Consultar o sumário
      Encontrar a página desejada
        Ler a página
      Recolocar o manual na caixa
  Começar a digitar a lista
    Digitar [CTRL]-[R] para promover o item da
      lista a novo cabeçalho
      Digitar sublista
    Digitar [CTRL]-[C] para voltar à lista anterior
  Editar a lista
    Digitar [CTRL]-[A] para alterar uma entrada
```

Também é possível editar as listas fora do BrainStorm. O WordStar e outros programas de processamento de texto as reconhecerão como documento se estiverem gravadas em disquetes. Nesse caso, podem-se editá-las, imprimi-las e, se necessário, regravá-las. Significa, por exemplo, a possibilidade de empregar o BrainStorm para preparar a sinopse de um livro, que seria escrito então com um processador de texto comum, acessando-se o arquivo BrainStorm.Doc enquanto o livro avança.

Devido à flexibilidade do programa, esse processo pode começar com as primeiras anotações de idéias básicas — mais uma vez, algo normalmente feito em pedaços de papel —, que então se tornam títulos de capítulos, sob os quais se escreve a história. Se um item de um capítulo crescer de forma a merecer transformar-se em capítulo à parte, isso será feito com facilidade. Da mesma forma, títulos de capítulos que acabam se revelando menos importantes do que se imaginava podem ser "rebaixados" a subitens.

Assim, o esboço rudimentar do livro começa a configurar-se e, uma vez que o WordStar e pacotes semelhantes podem acessar os arquivos do BrainStorm, a transformação desse esboço inicial no manuscrito acabado flui naturalmente, a partir dos primeiros processos de reflexão. Isso também quer dizer que não é necessário adotar os procedimentos trabalhosos exigidos pelo BrainStorm se se deseja, por exemplo, imprimir com espaço duplo ou triplo. Douglas Adams, autor de *The hitchiker's guide to the galaxy (Guia do caminhante na galáxia),* usou o BrainStorm dessa maneira para desenvolver o jogo de aventura baseado em seus livros.

O programa vem com um exemplo, que inclui a estrutura de uma agenda, um arquivo de nomes e endereços, uma lista de tarefas (com as sublistas "urgente", "importante" e "não esquecer"), além de uma seção para anotações. É de muito valor, podendo ser anexado aos modelos do usuário usando-se a opção Merge.

Aprende-se facilmente a usar o BrainStorm. O manual é muito claro (foi escrito com o próprio BrainStorm), dispensando recorrer-se a ele constantemente, uma vez que os comandos estão sempre à disposição num menu de tela.

Nem todos os comandos são mnemônicos, de forma que alguns mostram-se difíceis de memorizar. Porém o BrainStorm inclui o programa Installb, muito simples de usar, que permite reconfigurar todos os comandos e alterar todos os menus para adequação ao novo esquema determinado pelo usuário.

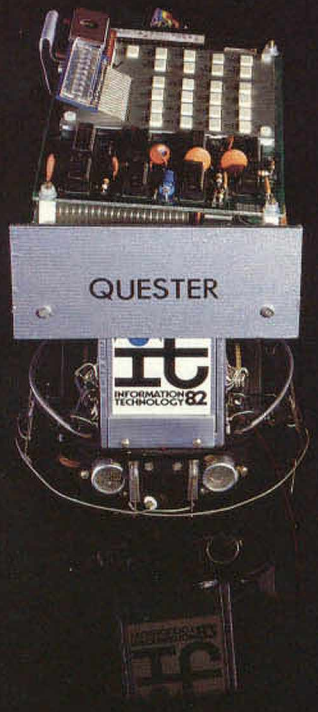
Listagem de porte

O BrainStorm combina um processador de texto com recursos de manipulação de listas encontrados nas linguagens LOGO e LISP. Qualquer item de uma lista pode tornar-se o cabeçalho de uma sublista, o que facilita o esboço e a organização de tarefas hierárquicas. No caso de cabeçalhos homônimos de sublistas, o BrainStorm manterá apenas uma lista desse nome e a acessará sempre que o homônimo for chamado, independentemente da lista em que for encontrado.

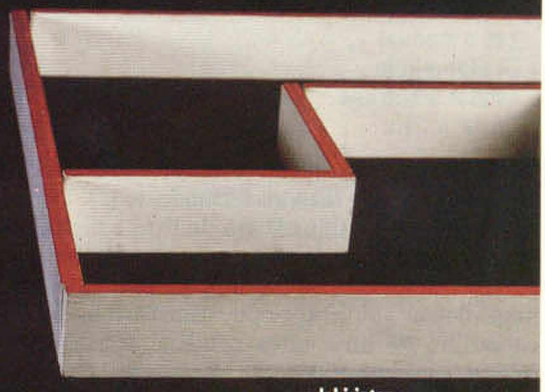
```
LISTA 1
GRANDES FILMES
1) Brasil
2) EUA
3) Itália
4) Japão

LISTA 1.1
BRASIL
1) Vencedores de festivais
2) Documentários
3) Co-produções

LISTA 1.2
EUA
1) Dramas
2) Comédias
3) Guerra

LISTA 1.3
ITÁLIA
1) Neo-realismo
2) Comédias

LISTA 1.4
JAPÃO
1) Dramas modernos
2) Samurais

HOMÔNIMO 1
VENCEDORES DE FESTIVAIS
1980 - Gaijin
1980 - Tudo bem
1981 - Pixote
1982 - Asa branca
1982 - Das tripas coração
1983 - Pra frente Brasil
1983 - Raoni
```

ROBÔS NA TRILHA

Robôs e labirintos



Robôs...
As competições de Micro Mouse (em que mouses-robôs fazem o possível para vencer um labirinto) são uma valiosa fonte de conhecimento prático e de habilidade técnica para os aficionados da robótica.
O Quester, de David Buckley, possui uma gama completa de sensores: ópticos, sônicos e sensíveis ao toque.



... e labirintos
Cada Micro Mouse possui um período de prática para "aprender" o esquema do labirinto, utilizando qualquer método que não exija comunicação externa.
Em seguida, deve atravessar o labirinto, numa luta contra o relógio — o objetivo básico é atingir o centro no menor tempo possível.

Existem três métodos principais para conseguirmos movimentar um robô, e os motores mais usados são os elétricos. Mas, se ele deve fazer aquilo que queremos, como controlar seus movimentos?

O método mais simples de movimentarmos um robô consiste num dispositivo mecânico que "lê" um cartão nele inserido, criado especialmente para essa tarefa. O perfil do cartão é seguido por uma came que comanda as alavancas responsáveis pelo direcionamento do robô. No passado, chegaram a ser construídas miniaturas de carros e robôs de brinquedo que funcionavam segundo esse princípio. Criava-se um programa utilizando uma tesoura para cortar o cartão na forma desejada, e o robô se movimentava conforme a margem recortada.

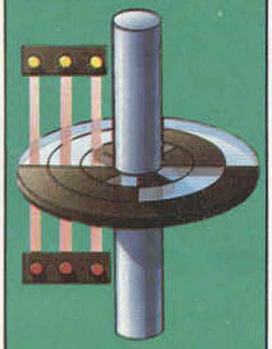
Outros robôs eram dotados de dispositivos que lhes permitiam seguir uma direção graças a relés eletromecânicos internos. Esses métodos, porém, tinham aplicações limitadas, pois as partes mecânicas costumam ser caras e um tanto imprecisas. Apesar disso, forneceram as bases para os métodos contemporâneos.

Um dos melhores meios hoje empregados consiste numa trilha traçada no solo para o robô seguir. Trata-se de um sistema semelhante ao usado para miniaturas de carros de corrida, que possuem um pino-guia inserido numa fenda contínua na pista. São dois os tipos mais comuns de robôs que seguem trilhas: os orientados por uma linha traçada no chão e os guiados por um fio.

Para seguir uma linha, o robô utiliza um elemento fotossensível (célula fotoelétrica ou sensor infravermelho), que determina se ele está numa área "iluminada" ou numa área "escura". Por exemplo, se a cor do chão for escura e a linha clara, a saída do sensor terá intensidade máxima toda vez que ele estiver sobre a linha. Assim, seguindo o caminho que proporciona essa máxima intensidade, o robô nunca se desviará da linha.

Mas essa técnica apresenta um problema: o que deve fazer o robô quando a saída do sensor

Codificador de eixo



O codificador de eixo permite que um robô saiba a distância que percorreu medindo o quanto giraram os eixos de suas rodas. O dispositivo consiste num disco calibrado adaptado a um eixo. A placa circular tem vários anéis concêntricos, cada um traçado em áreas transparentes ou opacas. A posição do eixo é determinada por uma fonte de luz e um fotossensor.

A precisão do codificador de eixo depende do número de anéis. A ilustração mostra um com três anéis, capaz de codificar os números binários de 000 a 111 (decimais de 0 a 7). Esse codificador oferece uma precisão de 360/8 = 45 graus. Oito anéis ofereceriam 1,41 grau.

sofre uma queda, indicando que ele abandonou a linha? Com um sistema único, o máximo que pode fazer é vagar de um lado para outro até que a saída do sensor retorne aos níveis anteriores, indicando que ele voltou à linha. Esse método não se verifica tão aleatório quanto parece. Por exemplo: se o robô se dirigia para a esquerda no momento em que a queda foi acusada, é razoável que ele vá para a direita a fim de reencontrar a linha. E, se a reencontrar, é razoável também que ele entenda estar a nova direção a seguir entre a rota que seguia antes (esquerda) e a que teve de seguir depois (direita).

Um sistema para reduzir o tempo que um robô "descarrilado" leva para reencontrar a direção certa faz uso de dois sensores, cada qual apontado para um dos lados da linha. Isto é: quando o robô anda na linha, a saída de ambos os sensores será baixa. Se começar a se desviar, a saída de um dos sensores acusará. Em outras palavras: ele saberá imediatamente que andou errado e em que direção. Se se desviar para a direita, o elemento esquerdo apontará, e o robô interpretará isso como um sinal para retornar à esquerda, o que o colocará de novo no caminho certo.

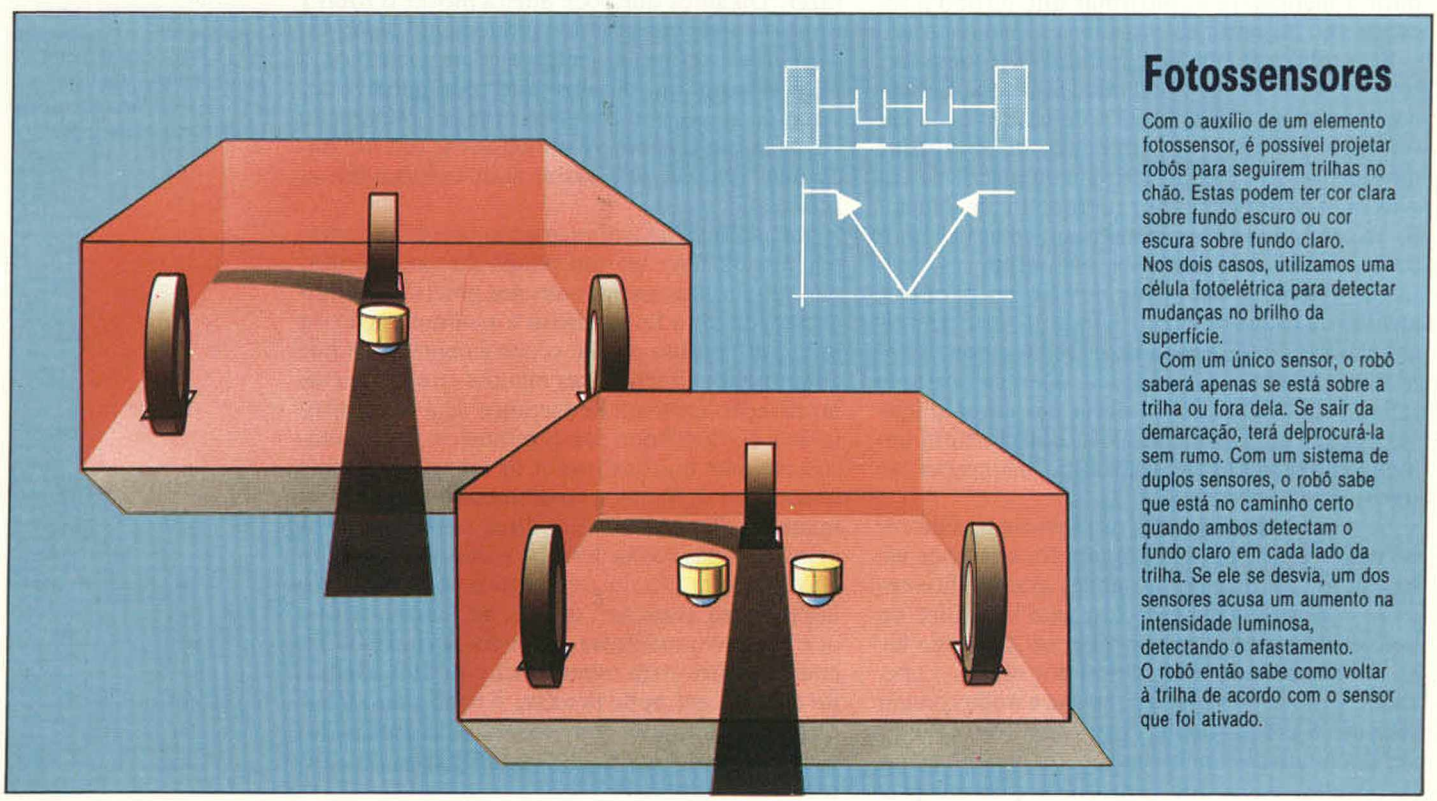
Para esse sistema, não é indispensável uma linha branca contra um fundo escuro, pois o contrário funciona da mesma forma. O importante é o contraste, e também que o programa instrua o robô quanto ao que fazer quando um sensor ler valores incorretos.

Outro sistema empregado em robôs que seguem trilhas consiste numa corrente elétrica percorrendo um fio estendido no chão. Essa corrente gera ao redor do fio um campo magnético, o qual é detectado pelo sensor. E este não precisa ser, necessariamente, complicado: uma pequena bobina de fio é suficiente para captar o campo magnético. Sob a ação do campo, resulta na bobina uma tensão, que, amplificada, atua como os fotossensores. Robôs industriais que se movimentam em várias direções são em geral orientados por um fio embutido no chão. Se seguissem uma linha pintada, tudo correria bem enquanto o chão não ficasse sujo.

Controle remoto

Outro método envolve um operador humano controlando o robô a distância. É particularmente útil quando as tarefas confiadas ao robô são desempenhadas em ambiente hostil ou perigoso ao ser humano. Exemplos: desativação de bombas; manipulação de produtos químicos perigosos ou materiais radioativos; trabalho em locais muito quentes ou muito frios, ou que, simplesmente, ofereçam grandes riscos. Um famoso robô desse tipo foi o soviético Lunokhod 1, levado à Lua pela nave espacial Luna 16, em 1970. Era um robô sobre rodas que colheu informações sobre o solo lunar, controlado por ondas de rádio enviadas da Terra.

O comando de robôs como esse não difere muito do de um aeromodelo radiocontrolado. Há dois tipos de sinais de rádio: o analógico, que varia de intensidade conforme a quantidade de movimentos exigidos do robô; e o digital, configurador de um padrão de bits que dá detalhes dos movimentos a serem executados. As comunicações analógicas em geral têm menos êxito que as digitais. Isso porque vários fatores podem interferir na intensidade dos sinais numa trans-



Fotossensores

Com o auxílio de um elemento fotossensor, é possível projetar robôs para seguirem trilhas no chão. Estas podem ter cor clara sobre fundo escuro ou cor escura sobre fundo claro. Nos dois casos, utilizamos uma célula fotoelétrica para detectar mudanças no brilho da superfície.

Com um único sensor, o robô saberá apenas se está sobre a trilha ou fora dela. Se sair da demarcação, terá de procurá-la sem rumo. Com um sistema de duplos sensores, o robô sabe que está no caminho certo quando ambos detectam o fundo claro em cada lado da trilha. Se ele se desvia, um dos sensores acusa um aumento na intensidade luminosa, detectando o afastamento. O robô então sabe como voltar à trilha de acordo com o sensor que foi ativado.

missão analógica. Ouça uma emissora de rádio distante e perceba como a recepção varia conforme a hora do dia e as condições atmosféricas. Pois o mesmo tipo de problema pode afetar as comunicações com os robôs.

Os métodos digitais também apresentam dificuldades, principalmente quando uma interferência provoca perda de bits ou insere bits em posições indevidas. Para evitar isso e operar com maior segurança, repetem-se as mensagens ao robô, e ele só começa a agir depois de receber várias vezes a mesma mensagem.

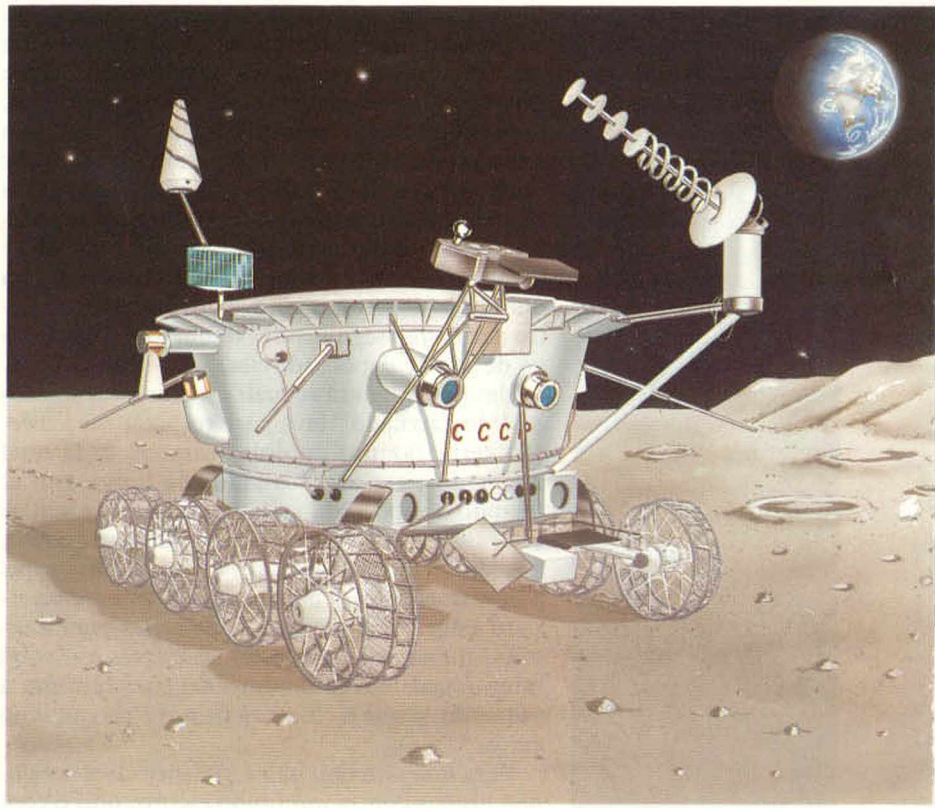
Sistemas de realimentação

Uma técnica mais sofisticada consiste em utilizar um sistema de loop (circuito completo), no qual o robô dá um retorno ao transmissor das informações que acabou de receber. Seria uma espécie de diálogo entre transmissor e robô. Por exemplo, o transmissor manda: "Mova-se para a frente"; ao receber essa mensagem, o robô pergunta: "Você disse 'Mova-se para a frente'?"; o transmissor responde: "Sim"; só então o robô se move. Isso evita erros desastrosos quando, digamos, o robô manipula lixo atômico ou está prestes a despencar para dentro de uma cratera na Lua.

Essas mesmas técnicas gerais aplicam-se a outros tipos de controle remoto. Por exemplo: certos robôs podem ser controlados a distância por emissores infravermelhos do tipo utilizado no controle remoto de aparelhos de tevê; por ultrasons ou por um som de natureza inconfundível, como o de assobios ou palmas. Seja qual for o método empregado, as técnicas básicas de transmitir a mensagem e confirmar que o robô a recebeu são as mesmas.

Se o robô trabalha bem próximo do operador, tornam-se desnecessárias técnicas tão sofisticadas: podem-se transmitir as ordens através de um fio. Existe ainda a possibilidade de se utilizar mais de um fio, o equivalente a uma série de canais num aeromodelo por radiocontrole. No caso do robô, porém, em geral usam-se os fios para proporcionar comunicações, não seriais, mas paralelas (envia-se um string de bits em paralelo, através de todos os fios, e não uma série de pulsos através de um único fio). Desse modo, a comunicação se faz mais rápida. Mais importante talvez seja o fato de a maioria dos computadores possuir uma saída paralela, que constitui um meio conveniente de transmitir instruções ao robô.

Se o operador controla os movimentos do robô pelo teclado de um computador e se pode vê-lo, então, em princípio, não há grande diferença se o controle for exercido pelo operador ou pelo computador. Isso porque, tal como no caso do aeromodelo controlado por rádio, o operador sempre vê o que o robô faz e pode corrigir seus erros prontamente. Se, porém, o robô estiver a alguma distância (na Lua ou mesmo na sala vizinha) ou se for controlado por um programa de computador e não por comandos introduzidos pelo teclado em tempo real, então ele terá de ser um pouquinho mais inteligente. Precisará, essencialmente, de alguma forma de realimentação (feedback).



Um passo gigantesco
Em 1970, o Lunokhod 1 pousou na Lua com a missão de colher informações sobre a superfície do solo. Embora não fosse um autêntico robô, ele pôde transportar uma enorme carga científica.

Como qualquer objeto espacial sob controle remoto, o Lunokhod tinha a desvantagem de um atraso de 3 segundos entre sua transmissão de informações à Terra e o retorno de um comando em resposta.

Esse processo capacita-o a ajustar o que está fazendo àquilo que já fez e com o que deveria fazer. Digamos que você queira mover o robô 1 m; se o estiver controlando diretamente, você poderá iniciar o movimento, avaliar o avanço e pará-lo assim que o metro tiver sido percorrido. Isso ocorre porque você tem uma realimentação visual sobre o robô: vê até onde ele foi, até onde quer que ele vá e assim pode corrigir-lhe as ações.

Na ausência de realimentação sensorial humana, o robô deve ter um pouco da sua própria, para poder movimentar-se com precisão. O tipo que segue uma linha utiliza a realimentação da linha; do mesmo modo, o que é controlado por computador precisa usar alguma forma de realimentação para que seu avanço seja exatamente de 1 m. O método mais comum de realimentação consiste no codificador de eixo, que associa as rotações dessa peça ao deslocamento linear da máquina. Trata-se de um disco preso aos eixos principais das rodas do robô e que mede quanto elas rodaram. Assim, quando o computador o instrui para avançar 1 m, ele, ao mesmo tempo que se move, monitora os sinais enviados pelos codificadores para saber quanto andou. Depois de avançar até onde precisa, o robô pára. Caso ultrapasse o limite, retrocederá à medida certa, calculada com base nas informações enviadas pelos codificadores.

WREN EXECUTIVE

Portátil, com duas unidades de disquete, modem de chamada automática incorporado e o BASIC da BBC, o Wren Executive volta, depois de alguns tropeços, a procurar seu espaço no mercado.

O computador Wren Executive tem uma história acidentada. Construído pela firma inglesa Thorn EMI, era comercializado originalmente pela Prism. Quando esta abriu falência, no início de 1985, parecia que o Wren seria mais uma vítima da turbulenta indústria de informática. Mais tarde, porém, a Opus Supplies encarregou-se da distribuição do computador, planejando relançá-lo. Este artigo analisa as possibiljdades da máquina no competitivo mercado profissional.

O Wren Executive foi concebido como máquina portátil, ou seja, um equipamento autosuficiente constituído por uma CPU, um monitor de vídeo e duas unidades de discos. Na parte traseira da máquina há uma alça de transporte. Como muitos outros portáteis, o Wren mostra-se um tanto pesado para transporte por longas distâncias e talvez seja mais adequado como computador de mesa, que pode ser eventualmente mudado de uma área de trabalho para outra.

O teclado é de configuração QWERTY e apresenta como única diferença os símbolos * e #, que têm suas próprias teclas. À esquerda, encontram-se cinco teclas de função programáveis — operadas em conjunto com a [Control] e a [Shift], elas proporcionam até quinze funções diferentes. Essas teclas possuem diversas finalidades, dependendo da aplicação, embora seu uso predominante se restrinja aos comandos CP/M simples como PIP e RENAME. À direita estão as quatro teclas de movimentação do cursor e a [Home], para retorná-lo ao canto superior da tela.

Medindo 150 x 105 mm, com resolução de texto de 80 x 25 caracteres, sobre fundo âmbar (em lugar do verde usual), a tela favorece longos períodos de trabalho. À direita ficam as duas unidades de discos flexíveis de 5 1/4 polegadas.

Um dos problemas do vídeo é o desenho do gabinete. Além de sua pequena altura, o equipamento não possui pés reguláveis para inclinação, dificultando a visualização da tela. Esta fica embutida no gabinete, e tem sua parte superior um tanto obscurecida por uma aba de 40 mm que se projeta sobre ela. Embora não se trate de um problema grave, traz algum incômodo ao usuário.

O Wren é bem provido de interfaces. Há várias saídas na parte traseira, permitindo sua conexão com uma ampla variedade de periféricos. Diferentemente de muitas máquinas comerciais na mesma faixa de preço, o Wren incorpora um modem de chamada automática. Seu proprietário, portanto, pode acessar todas as principais bases de dados sem precisar adquirir qualquer outro equipamento. Acompanha o Wren um cabo para ligação ao sistema telefônico. O software de comunicações fornecido com o equipamento permite armazenar vários números de telefone e chamá-los automaticamente quando necessário.

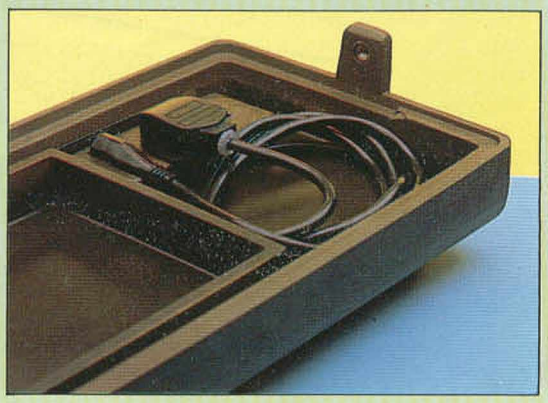
Ao lado da saída de comunicações existem duas outras, destinadas aos conectores de controles externos, por exemplo, de mouses e joysticks. Ao lado encontra-se um conector para saídas seriais de comunicações RS232C, que costumam ser usadas em computadores profissionais para modems externos. Como tal utilização não é uma necessidade no Wren, essas saídas são mais empregadas para redes locais e impressoras seriais.

Uma interface paralela padrão Winchester permite conexão a uma unidade de discos rígidos, que acrescenta de 0,5 a 50 Mbytes à capacidade de armazenamento. Há também uma interface paralela para impressoras compatível com o padrão Centronics. A máquina dispõe também de saída para monitor colorido RGB. Entre as interfaces para impressora e para monitor, há um diminuto botão [Reset], para a reinicialização do sistema.

Pássaro incomum
O Wren Executive é uma máquina portátil barata, com a característica incomum de rodar o BASIC da BBC, assim como o sistema operacional CP/M. Entre outros recursos, há duas unidades de discos, monitor monocromático e modem embutido.

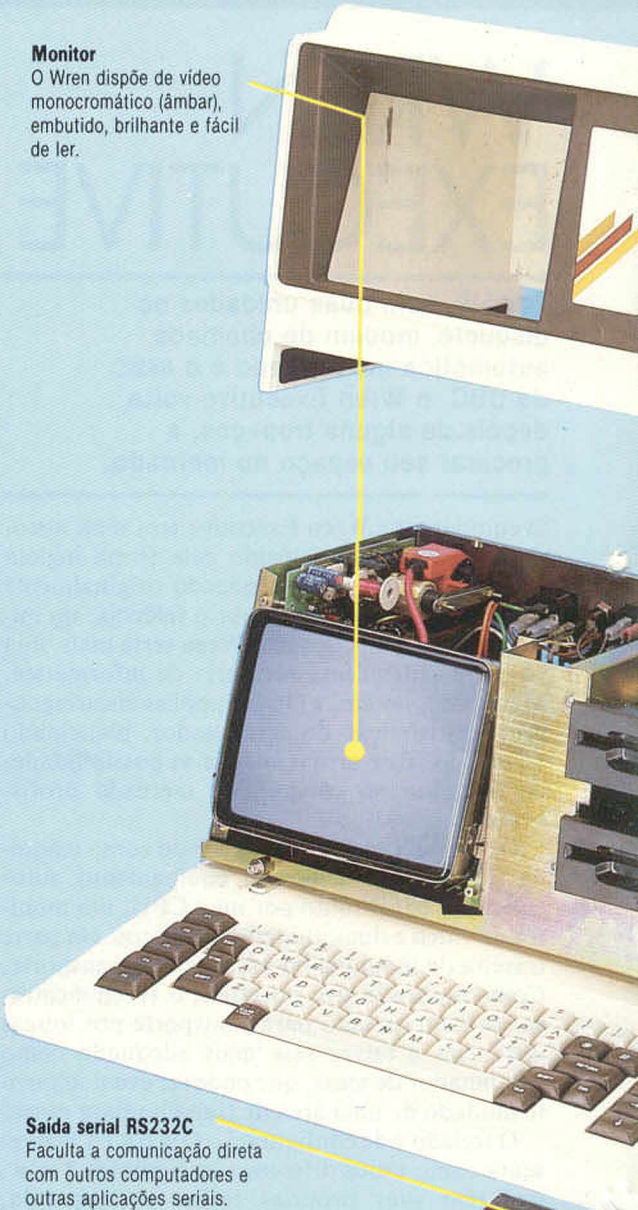
Riqueza de interfaces
Uma das melhores caracteristicas do Wren é a quantidade de interfaces existentes na configuração padrão. Além das interfaces comuns, Centronics e RS232C, há também uma para linha telefônica com modem incorporado. Existe, ainda, uma saida paralela para unidade de discos rigidos tipo Winchester.

Fundo útil
O monitor e as unidades de discos são protegidos, durante o transporte, por uma tampa de plástico reforçado, que se encaixa à parte frontal da máquina. Seguindo a filosofia "ligar na tomada e sair rodando", o fabricante tomou a atitude incomum de fornecer um cabo de alimentação com a tomada já encaixada. Esse componente fica acondicionado num compartimento da tampa durante o transporte.



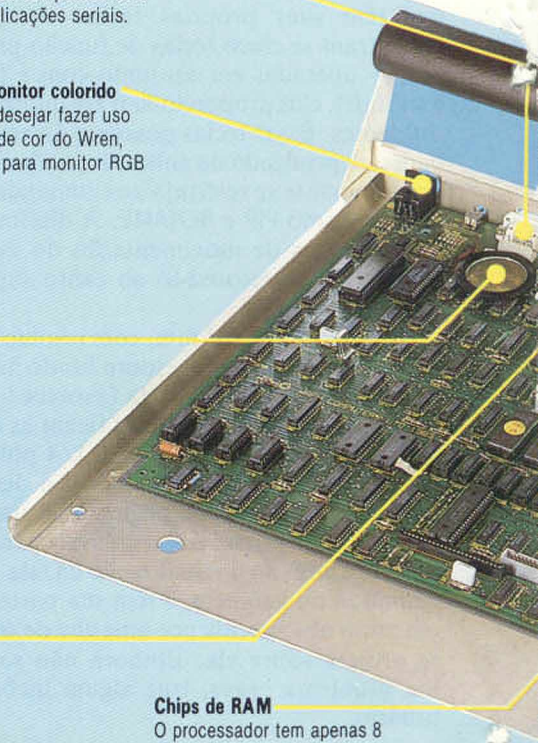
Promessa de perfeição
O Wren é fornecido com seu próprio conjunto de softwares — Perfect Writer, Perfect Filer e Perfect Calc. Também há um programa genérico, o Executive Desktop, um software de comunicações, e o BASIC da BBC.

Monitor
O Wren dispõe de vídeo monocromático (âmbar), embutido, brilhante e fácil de ler.



Saída serial RS232C
Faculta a comunicação direta com outros computadores e outras aplicações seriais.

Saída para monitor colorido
Se o usuário desejar fazer uso dos recursos de cor do Wren, há uma saída para monitor RGB externo.



Alto-falante
Os recursos de som provêm do alto-falante do próprio Wren.

Interface para discos rígidos
Conecta uma unidade de discos rígidos tipo Winchester, ampliando bastante a capacidade de armazenamento do computador.

Chips de RAM
O processador tem apenas 8 bits, mas o Wren endereça até 400 Kbytes de RAM.

O Wren baseia-se no conhecido microprocessador de 8 bits Z80B. Talvez seja esse o ponto mais fraco do computador. No momento em que os fabricantes de micros domésticos competem furiosamente no desenvolvimento de máquinas de 16 bits, o Wren parece um tanto antiquado em seu processamento. Embora de rapidez razoável para os padrões de 8 bits, seu desempenho não se compara ao dos micros de 16 bits, como o IBM PC e o Macintosh da Apple.

Como o Wren faz uso do processador Z80, era natural que seus idealizadores decidissem pelo CP/M como sistema operacional. Pelos padrões recentes, no entanto, o CP/M deixa muito a desejar quanto à interação com o usuário, apesar de sua potência. Os projetistas minoraram esse problema incorporando ao sistema operacional um procedimento comandado por menus, o que permite a escolha de um aplicativo por meio do cursor e da tecla [Return]. O sistema, então, espera a inserção do disco apropriado e executa o aplicativo de modo automático. Assim, muitos perceberão que não há necessidade de usar diretamente o CP/M, já que quase tudo o que o usuário comercial médio precisa está incluído no

Os programas fornecidos com o computador são o sistema CP/M, o processador de texto Perfect Writer, o Executive Desktop (um pacote para controle de agendas — datas, compromissos etc.), o banco de dados Perfect Filer, a planilha financeira Perfect Calc e um disco contendo software de comunicações e o BASIC. Embora a máquina tenha uma implementação quase total da linguagem, por se basear no processador Z80 e não no 6502, surgem algumas diferenças, principalmente nas rotinas de entrada e saída. A implementação da linguagem também parece ter gerado alguns erros. Por exemplo, nos modos 0 e 1, o cursor desaparece da tela. A tecla [Delete] não funciona nesses modos, e o retrocesso do cursor bem como a tentativa de reescrita resultam na superposição do novo caractere ao anterior. Isso impede uma programação eficiente nesses modos, embora os outros pareçam funcionar bem.

Em última análise, o sucesso do Wren Executive depende de ele encontrar seu espaço correto no mercado. Com modem embutido, sistema operacional CP/M, BASIC da BBC e um custo baixo, o Wren parecia valioso quando de seu lançamento, no primeiro semestre de 1984. Mas o potencial de vendas das máquinas profissionais de 8 bits, mesmo na faixa inferior do mercado, diminuiu muito de lá para cá. Além disso, a máquina ainda é muito cara para atacar seriamente o mercado dos micros domésticos.

Sobra então o mercado educacional, onde seu emprego pode ser maior. O uso do BASIC da BBC, de ampla utilização nas escolas britânicas, combinado ao modem embutido — capaz de acessar as várias bases de dados que vão se tornando disponíveis —, poderia interessar os estabelecimentos educacionais que tentam obter o máximo retorno para seus investimentos.

WREN EXECUTIVE

MICROPROCESSADOR

Z80B.

CLOCK

6 MHz.

SISTEMA OPERACIONAL

CP/M.

VÍDEO

Modo texto: 80 x 25 caracteres; modo gráfico: 640 x 256 pixels.

TECLADO

No padrão QWERTY, com 57 teclas, sendo cinco de funções programáveis e cinco de controle do cursor.

LINGUAGENS DISPONÍVEIS

BASIC da BBC.

INTERFACES

Saída para monitor, interface padrão Centronics, saída serial RS232C, linha telefônica, duas saídas para paddles, interface para discos rígidos.

DOCUMENTAÇÃO

Bem apresentada, oferece ao usuário explicações completas. Além do manual do usuário, mais um para cada aplicativo fornecido.

AVENTURAS E SIMULAÇÕES

Bastante divertidos, os jogos baseados em estratégias que simulam situações da vida real testam nossa capacidade de planejamento, bem como outras, que empregamos no dia-a-dia.

Além de proporcionar jogos que exigem reflexos e movimentos rápidos para os fãs do "mata-marcianos", o computador é um excelente meio para um desafio diferente. Trata-se do jogo de simulação, que requer habilidades muito diferentes das exigidas pelos jogos tipo fliperama. A "simulação" consiste em criar na memória do microcomputador modelos da vida real para os propósitos do jogo. Esses modelos podem ser simuladores em "tempo real", como os programas de simulação de vôo, ou baseados em estratégias.

Nestes, o jogador, ou grupo de jogadores, recebe uma série de tarefas a realizar e algumas informações que o ajudam na tomada de decisões. Um exemplo típico é o clássico Banco Imobiliário, no qual os participantes recebem dinheiro com o objetivo de aumentar seus lucros. O que distingue um jogo de simulação dos de aventura e ação é que, neste último, o jogador encontra armadilhas e outros riscos sem saber exatamente quando surgem; já numa simulação, em geral ele pode prever o que está por vir, direcionando seus recursos para vencer contingências esperadas.

Esse tipo de jogo, portanto, exige de seus participantes capacidade de organização e previsão; em outras palavras, habilidade de planejar com antecedência, em vez de decidir rápido sobre uma situação imediata. Para refletir o mundo real, nem tudo num jogo de simulação deve ser deixado correr exatamente como planejado. Uma boa simulação sempre introduz fatores aleatórios (na verdade, pseudo-aleatórios) para frustrar os esquemas do participante. Um bom jogador deve, portanto, considerar o fator "acaso" a fim de minimizar os danos causados por eventos não previstos.

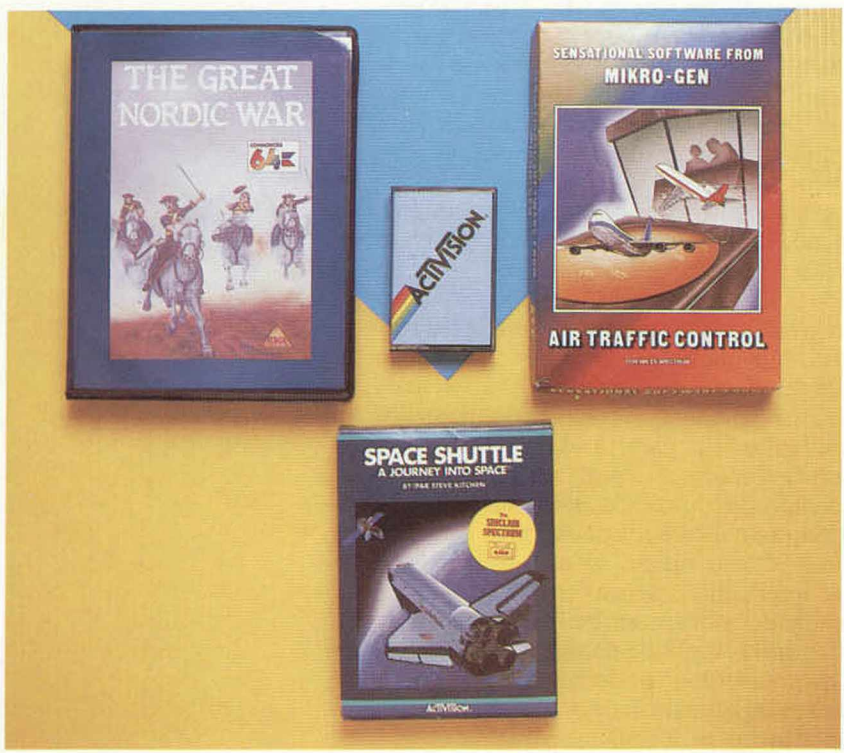
Os jogos de simulação têm a vantagem de permitir que as pessoas joguem em equipe, combinando inteligência e poder de decisão contra o computador, para cumprir uma tarefa predeterminada. Por essa razão, esses jogos começam a encontrar larga aceitação em diversas escolas, pois possibilitam o aperfeiçoamento do tipo de comportamento que se espera dos alunos no mundo adulto do trabalho, combinando recursos e talentos de um pequeno grupo para atingir um certo objetivo.

Além de todos esses atributos, os jogos de simulação projetam-se numa área raramente tocada pela maioria das aplicações computacionais: a das questões éticas e morais. Por exemplo, os jogadores podem ser levados a ponderar a respeito do lucro crescente contra o bem-estar dos empregados. Quando se tomam tais decisões em equipe, costumam ser precedidas de debates acalorados sobre o certo e o errado.

Muitas situações diferentes prestam-se bem à simulação em computador. Geralmente o fator comum está em envolverem a realização de uma determinada tarefa, a fim de se atingir um objetivo específico. Em certos casos, este pode ser escolhido pelo jogador. A meta consiste em permanecer nos negócios por um certo período de tempo ou, alternativamente, ganhar um bilhão de cruzeiros. Muitas simulações se baseiam em estratégias financeiras, mas outras têm objetivos mais filantrópicos.

Um jogo de simulação atualmente utilizado nas escolas primárias inglesas constitui um projeto para localizar e recuperar os destroços do galeão elizabetano *Mary Rose*. Os jogadores recebem também documentos que complementam o programa e dão pistas adicionais que ajudam no processo de tomada de decisões. Num outro jogo, de simulação mercantil, os participantes têm de equipar e despachar um navio para o No-

Mundos imaginários
Abaixo estão três exemplos dos principais tipos de jogos de simulação. O Space Shuttle é uma simulação realista de uma viagem do ônibus espacial. No Air Traffic Control, uma simulação em tempo real, você dá instruções a vários aviões para que possam decolar e aterrissar com segurança. E The Great Nordic War simula a Guerra dos Trinta Anos, com o jogador desempenhando o papel de Carlos XII da Suécia.

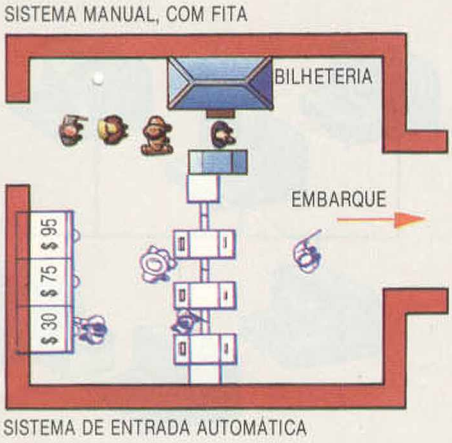


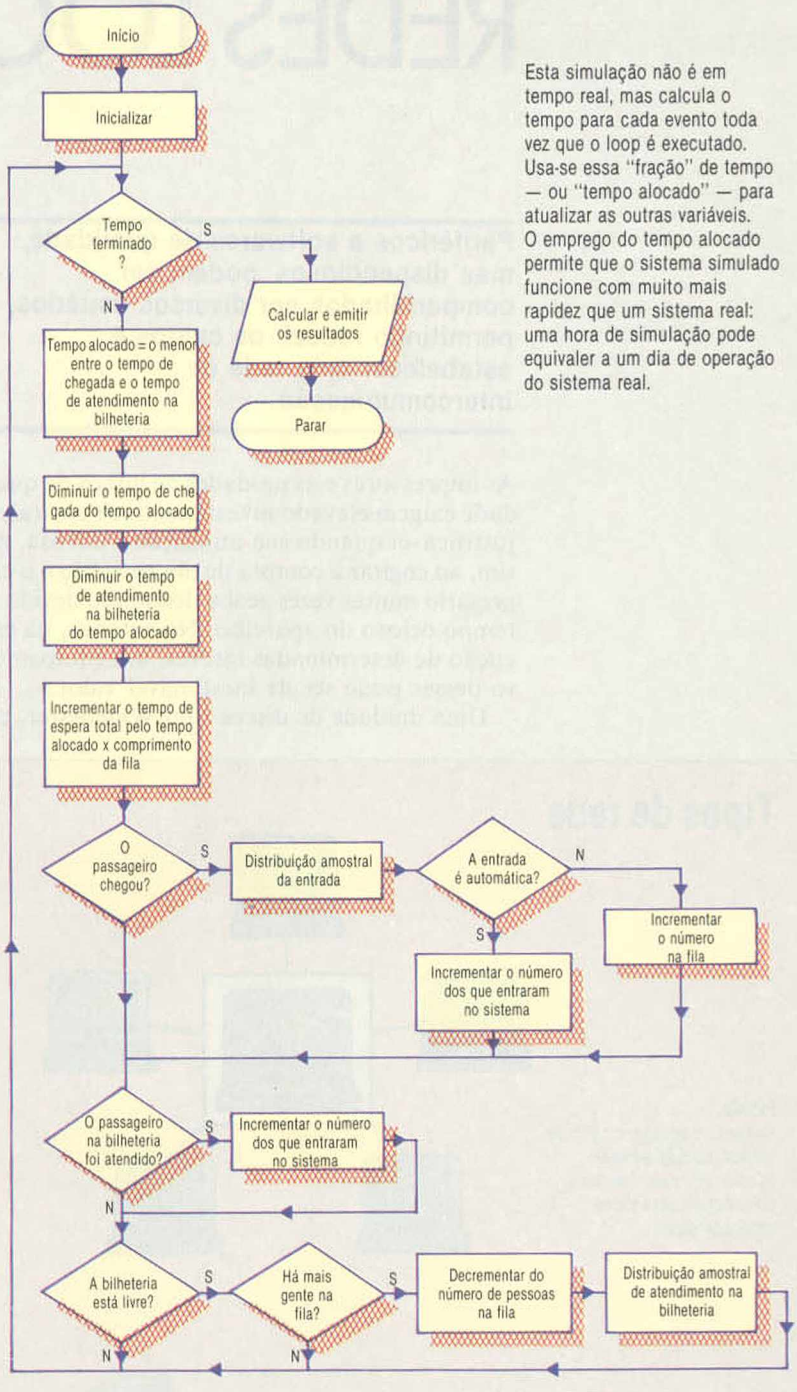
Simulação de serviços

As simulações têm grande utilidade na projeção do fluxo de tráfego e de sistemas de serviços, onde a natureza do modelo as torna ideais para uso em computadores. Tais simulações normalmente se baseiam em princípios estatísticos e matemáticos que prevêem distribuições de freqüências das variáveis usadas.

Os princípios de uma simulação podem ser demonstrados observando-se um sistema de serviços simples como o metrô. O esquema existente permite acesso à plataforma de embarque de duas maneiras: pagando diretamente na bilheteria, ou pelos bloqueios automáticos, que são operados por bilhetes comprados antecipadamente. O fluxograma ilustra os processos envolvidos na produção de uma simulação simples do sistema. São usadas três distribuições: uma para a chegada dos passageiros, outra para o tempo de atendimento na bilheteria e uma terceira para o método de entrada (manual ou automático).

As duas primeiras distribuições derivam de observações estatísticas e métodos matemáticos; a terceira variará a cada passagem do programa, para se observarem as ramificações das várias proporções de bloqueios manuais e automáticos. Essa simulação simples calculará o tempo médio de espera dos passageiros que entram no sistema.





Esta simulação não é em tempo real, mas calcula o tempo para cada evento toda vez que o loop é executado. Usa-se essa "fração" de tempo — ou "tempo alocado" — para atualizar as outras variáveis. O emprego do tempo alocado permite que o sistema simulado funcione com muito mais rapidez que um sistema real: uma hora de simulação pode equivaler a um dia de operação do sistema real.

vo Mundo, no século XV. Estruturado como uma série de módulos interligados, seu programa é elaborado de forma tal que o cenário seja facilmente alterado, proporcionando uma base para diferentes contextos. Assim, a simulação pode ser situada em outras épocas, transformando-se, por exemplo, numa jornada de ficção científica para incremento do comércio com um planeta distante.

O objetivo do jogo, para um ou mais jogadores, é comercializar bens para maximizar o lucro, contrabalançando esse propósito com a segurança e o conforto da tripulação. O jogo se divide em três partes: preparação, viagem e comércio. A primeira fase envolve a obtenção de recursos e o planejamento da viagem; a segunda introduz elementos aleatórios no jogo. A habilidade em lidar com essas contingências dependerá muito de como se executou o planejamento durante a primeira fase do jogo. A terceira fase envolve o estabelecimento de comércio lucrativo e de boas relações com os habitantes locais.

As principais tarefas do programa consistem em acompanhar as decisões dos jogadores, criar diversas situações imprevisíveis, analisar e monitorar a condição dos participantes conforme o desenrolar do jogo.

REDES LOCAIS

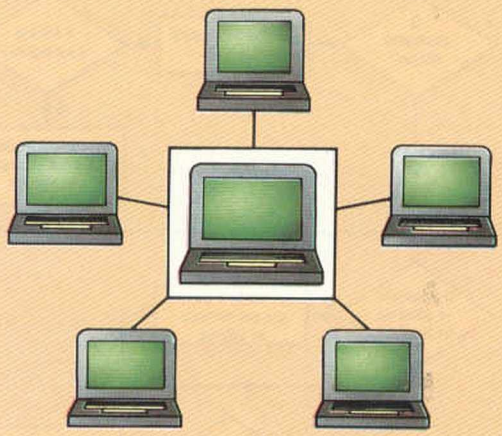
Periféricos e softwares de qualidade, mas dispendiosos, podem ser compartilhados por diversos usuários, permitindo reduzir os custos e estabelecer uma rede de intercomunicação.

As impressoras e as unidades de discos de qualidade exigem elevado investimento de capital, só justificável quando sua utilização é intensa. Assim, ao cogitar a compra de um periférico, o empresário muitas vezes acaba desistindo devido ao tempo ocioso do aparelho. No entanto, na execução de determinadas tarefas, um equipamento desses pode ser de inestimável valor.

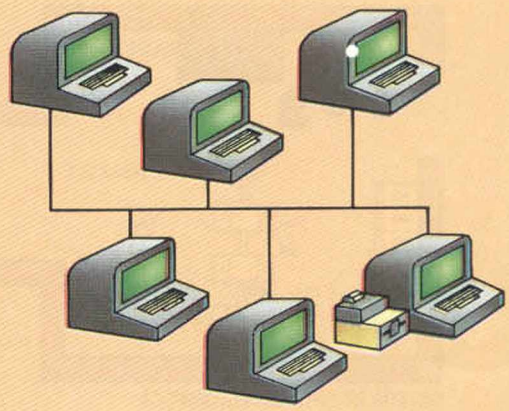
Uma unidade de discos tipo Winchester, por exemplo, revela-se de enorme importância em trabalhos que necessitam de acesso rápido à informação. Mas sua capacidade de memória pode ser excessiva para a maioria dos proprietários de micros. Esse problema se resolve pela implantação de redes de computadores, que permitem a um grupo de usuários compartilhar recursos de hardware, software e dados.

Esse uso conjunto de dados, quando efetuado através da interligação de computadores, estações, serviços, periféricos e congêneres numa área local, isto é, num mesmo escritório, prédio ou conjunto de prédios, origina a chamada rede local. Na medida em que organizações como bancos, indústrias, hospitais e escolas começam a aumentar seu número de computadores, torna-se necessária a interligação para compartilhar recursos e informações.

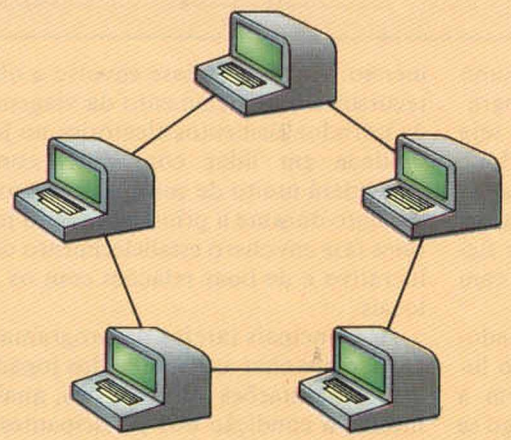
Tipos de rede



Estrela
Nesse tipo de rede, o controle central, ao qual se liga o operador de cada máquina, comanda os periféricos compartilhados.

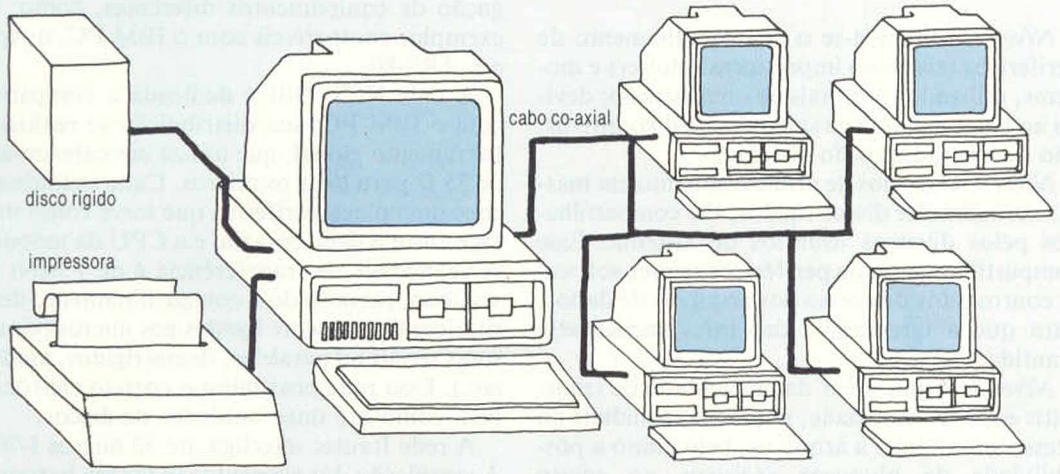


Barramento
Nessa disposição, os dados e as mensagens de controle passam diretamente de um usuário para outro.



Anel
Menos usada que as outras, essa configuração conecta os elementos em sequência. A informação chega ao destino após percorrer mais de metade da rede.

Degradação lógica



Para se ter uma idéia do que seja a degradação lógica num sistema de rede, consideremos diversos usuários compartilhando o mesmo banco de dados, representado por uma unidade de disquetes e uma impressora, ligados diretamente a um dos micros. Se, por exemplo, todos os micros necessitam buscar uma informação na unidade de disquete simultaneamente, a CPU do elemento que tem os periféricos precisa compartilhar seu tempo para atender às pesquisas. Isso resulta num tempo de espera em cada micro — é a degradação lógica.

O problema se mostra bem maior no caso de unidades de disquetes do que no de impressoras. No entanto, o sistema de acesso às informações pode ser aprimorado utilizando-se discos rígidos, que permitem reduzir o tempo de resposta.

A utilização das redes locais varia conforme a definição das necessidades. Por exemplo, em determinada empresa, a rede pode controlar o fluxo de caixa, estoque, folha de pagamento, emissão de notas fiscais e a contabilidade geral. Já em outra, pode servir para compartilhar dados oriundos de computadores de grande porte, processando-os independentemente.

Além da questão do custo dos periféricos, a interligação de equipamentos em rede estimula o desenvolvimento de todo o sistema sobre uma mesma linha de aplicativos, o que resulta em economia de software. As redes locais permitem ainda o aumento dos recursos físicos (periféricos) disponíveis para cada estação (usuário), além de maior interligação entre aplicações e confiabilidade elevada a baixo custo. Soma-se a isso a possibilidade de crescimento gradativo conforme as necessidades da organização.

Uma rede local executa algumas tarefas básicas, como, por exemplo:

a) Transferência de arquivos.

b) Teletipo — transmissão de mensagens de tamanho fixo (512 bytes).

c) Comunicação entre processamentos distintos.

d) Conversão de códigos de comunicação.

e) Detecção e recuperação de erros na transmissão.

f) Conversão de velocidade de transferência.

Para a realização dessas tarefas, são necessárias algumas rotinas escritas em linguagem de baixo nível (ASSEMBLY), que disciplinam o acesso das comunicações ao meio de transmissão.

O conjunto dessas rotinas chama-se protocolo. Os dois tipos de protocolo mais utilizados são o TOKEN, "passagem por permissão", e o CSMA (Carrier Sense Multiple Acess) "por contenção". No TOKEN, o dado transmitido é analisado pelas máquinas, numa de cada vez, e retransmitido às demais que compõem a rede quando seu endereço corresponde ao de outra máquina. No CSMA, ao contrário, todas as máquinas podem acessar o meio ao mesmo tempo e só receber a informação a elas destinada.

O que determina, na grande maioria das vezes, o tipo de protocolo é a disposição física da rede, ou seja, o tipo de interligação existente entre os equipamentos, também conhecida como arquitetura da rede. Os três tipos mais difundidos são a rede em anel (ring, ou loop), em barramento global (bus) e em estrela.

Todas essas considerações dizem respeito ao meio físico de interligação dos elementos da ma-

lha. No estudo das redes, levam-se em conta, ainda, as ligações lógicas possíveis entre os componentes, que se resumem em três níveis, explicados a seguir.

Nível 1. Objetiva-se o compartilhamento de periféricos tais como impressoras, plotters e modems, utilizados por mais de uma estação; devido ao baixo volume de saída desses dispositivos, não há degradação do sistema.

Nível 2. Os meios de armazenamento em massa, normalmente discos rígidos, são compartilhados pelos diversos usuários do sistema. Esse compartilhamento de periféricos requer software controlador de acesso aos arquivos de dados, para que a integridade das informações seja mantida.

Nível 3. Aqui, além da capacidade de transmitir em alta velocidade, requer-se facilidade no acesso simultâneo a arquivos, bem como a possibilidade de bloquear registros ao acesso (LOCK). Sem esse nível não é possível manter-se a integridade dos dados, eliminar a redundância de informações e consolidar dados produzidos por diferentes usuários.

Redes nacionais

Entre os fabricantes de redes para equipamentos nacionais destacam-se a Cetus, a Itautec e a NCT. A Cetus utiliza um nodo modelo CS-1000 (ou CS-1200) ou uma placa periférica (quando utilizada em compatíveis com o IBM PC). Tanto o nodo quanto a placa servem como interface física entre o meio (o par trançado) e o hardware distintos de cada máquina.

A ligação entre o nodo e o equipamento se faz via saída padrão RS232C. Os periféricos ligados a essa rede recebem também um nodo. No caso de unidades de discos rígidos, são utilizados os nodos CS-1200. A ligação de impressoras seriais também necessita de nodo.

A grande vantagem desta rede está na interligação de equipamentos diferentes, como, por exemplo, compatíveis com o IBM PC, o Apple e o TRS-80.

A rede NCT-7301 é dedicada a compatíveis com o IBM PC; sua distribuição se realiza em barramento global, que utiliza um cabo co-axial de 75 Ω para ligar os micros. Cada máquina recebe uma placa periférica que serve como interface entre o cabo co-axial e a CPU da máquina. A velocidade de transferência é de 1 Mbit por segundo, permitindo o compartilhamento de periféricos diretamente ligados aos micros (impressoras seriais ou paralelas, discos rígidos, modems etc.). Essa rede possibilita o correio eletrônico, bem como até doze unidades de discos.

A rede Itautec interliga até 32 micros I-7000. A vinculação dos elementos se faz via barramento global, por meio de cabos co-axiais, a distâncias de até 1.000 m entre o primeiro e o último micro.

Por ora, ela só permite utilização como o I-7000, embora já esteja programado um novo produto para ligar também micros de 16 bits, como o I-7000 PCxt.

São dois os tipos de estação utilizados por essa rede: micros usuários e micros servidores (onde estão conectados os periféricos que serão compartilhados pelos micros usuários). A única exigência é que exista pelo menos um servidor na rede.

O hardware da Itautec não utiliza a saída serial do micro. Uma placa de expansão instalada no equipamento permite velocidades de até 2,5 Mbits por segundo.

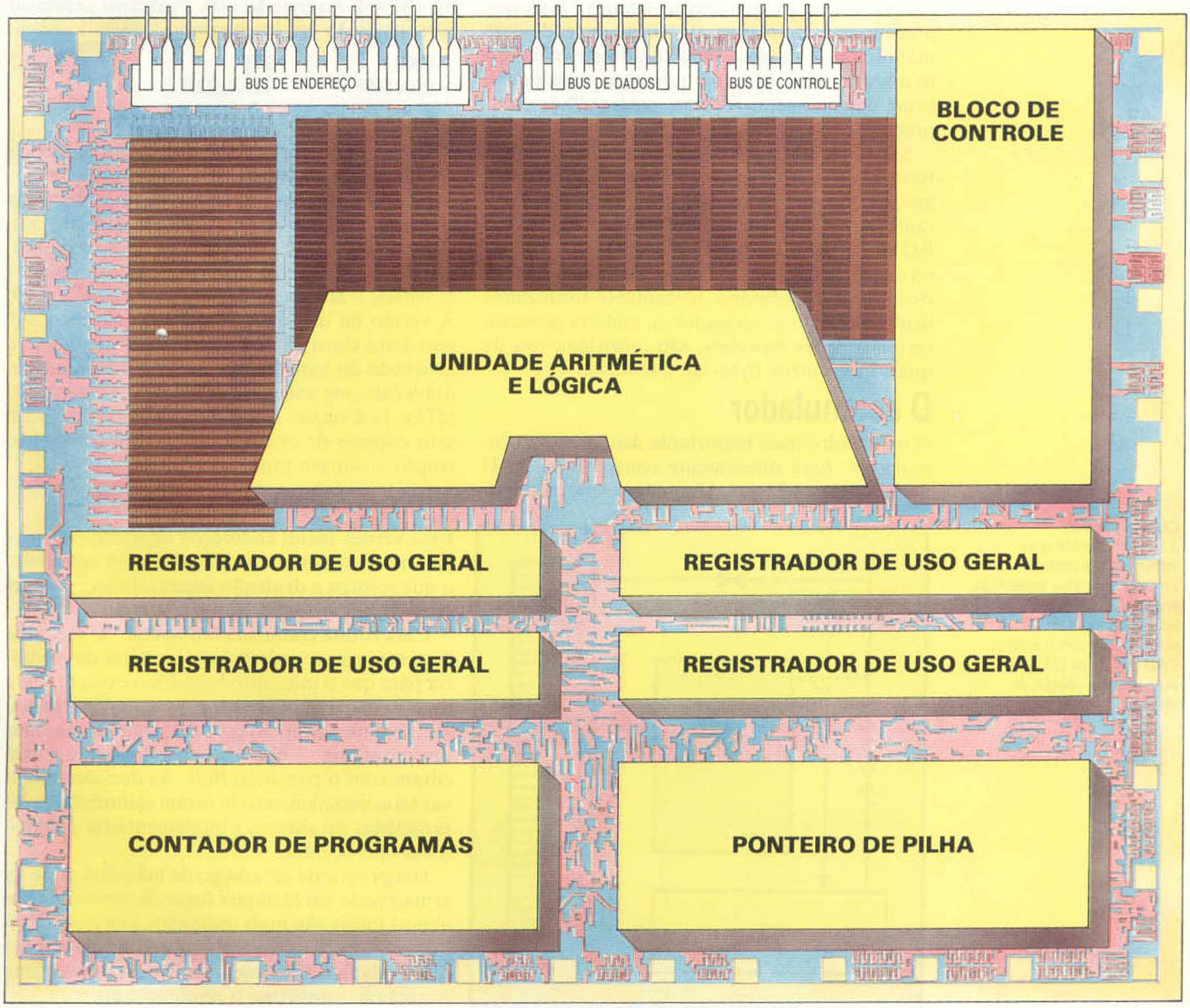
Fabricante	Tipo de rede	Meio físico de interligação	Compartilha	Desenvolvido para o equipamento
Cetus	Barramento	Par trançado ou co-axial	Discos, impressoras, discos rígidos, modem	Mistura qualquer equipamento
Itautec	Barramento	Cabo co-axial	Discos rígidos, impressoras	I-7000
NCT	Barramento	Cabo co-axial	Discos rígidos, impressoras, modems	Só compatíveis com IBM PC (16 bits)

BYTE A BYTE

Atividades de controle
O bloco de controle admite a instrução codificada (em binário), interpreta-a e induz todas as outras partes da CPU a atuar de modo adequado. Por exemplo, se uma instrução indica que o conteúdo do acumulador deve ser armazenado em determinada posição da memória, o bloco de controle coloca o endereço no bus de endereço, envia sinais instruindo a memória a armazenar dados, e coloca o conteúdo do acumulador no bus de dados.

Utilizando códigos mnemônicos e símbolos alfanuméricos, a linguagem ASSEMBLY é uma tradução direta do código de máquina, porém mais compreensível que este aos olhos humanos.

Já vimos como as linhas de um programa em BASIC são interpretadas e reduzidas a símbolos seguidos por dados ASCII. Vemos assim que o BASIC, embora considerado linguagem de alto nível, consiste apenas em seqüências de instruções; cada instrução, por sua vez, consiste numa palavra de comando seguida por dados. Os comandos são substituídos por símbolos que se encontram a apenas um nível acima dos códigos operacionais de máquina. Como as palavras de comando e seus dados (variáveis, números e strings) assemelham-se àquelas utilizadas na linguagem natural, e as instruções são separadas por números de linha ou por dois-pontos, um programa em BASIC nos parece de nível muito mais alto do que realmente é. Portanto, podemos concluir que o código de máquina precisa apenas de um pouco de tratamento visual para



se tornar compreensível a nossos olhos.

Esse tratamento visual é a linguagem ASSEMBLY. Nela, códigos alfabéticos mnemônicos (fáceis de memorizar) como LDA e ADC substituem os códigos de operação de um só byte, que o microprocessador entende, e utilizam-se símbolos alfanuméricos como LABEL1 e TFLAG, em vez de endereços de memória e dados numéricos. O microprocessador não compreende a linguagem ASSEMBLY; por isso, antes de executar um programa, devemos traduzi-lo para código de máquina — seja manualmente, pelo programador, seja por intermédio de um programa denominado assemblador, ou montador.

A linguagem ASSEMBLY é uma tradução direta do código de máquina. Substituindo os códigos mnemônicos por códigos de operação e os símbolos por números, convertemos um programa em código executável. Mas o ASSEMBLY revela-se muito mais compreensível aos olhos humanos do que o código de máquina, e por isso é útil na programação. Na prática, escrevemos programas em linguagem ASSEMBLY, e só nos preocupamos com as equivalências em código de máquina nos estágios finais do desenvolvimento do programa. Mas, no momento, vale a pena examinar ambas as formas, para melhor compreensão.

Sabemos que o microprocessador executa diferentes funções lógicas, mas, em essência, ele apenas manipula o conteúdo da memória. Para tanto, atua diretamente nos chips de RAM e de ROM ou trabalha com sua própria memória interna, os registradores. Estes consistem em vários bytes de memória fisicamente localizados dentro do microprocessador, e, embora possuam certas funções especiais, são indistinguíveis de quaisquer outros bytes de memória.

O acumulador

O registrador mais importante denomina-se acumulador. Está diretamente conectado à ALU (Arithmetic and Logic Unit, "unidade aritmética e lógica") e, assim, é usado com mais freqüência do que qualquer outro registrador. Para utilizá-lo, devemos introduzir as informações num processo denominado "carregar o acumulador". Usando a linguagem ASSEMBLY, dizemos que o 6502 é carregado por meio da operação LDA, e o Z80, pela operação LD A (ambas abreviações de LoaD Accumulator). Tão essencial quanto carregar o acumulador é dele extrair informações; a linguagem ASSEMBLY do 6502 utiliza para tanto a operação STA (STore the Accumulator, "armazenar o conteúdo do acumulador"). Para o Z80, no entanto, carregar e armazenar são casos diferentes de uma mesma operação: a transferência de dados. Assim, a extração de informações do acumulador também é efetuada pela operação LD A, mas num formato diferente.

Vamos supor, então, que queremos escrever um programa em linguagem ASSEMBLY para copiar o conteúdo de 1 byte de memória no byte de memória seguinte. Começamos copiando o byte $09FF no byte $0A00. Podemos expressar essa operação em linguagem ASSEMBLY assim:

6502	Z80
LDA $09FF	LD A,($09FF)
STA $0A00	LD ($0A00),A

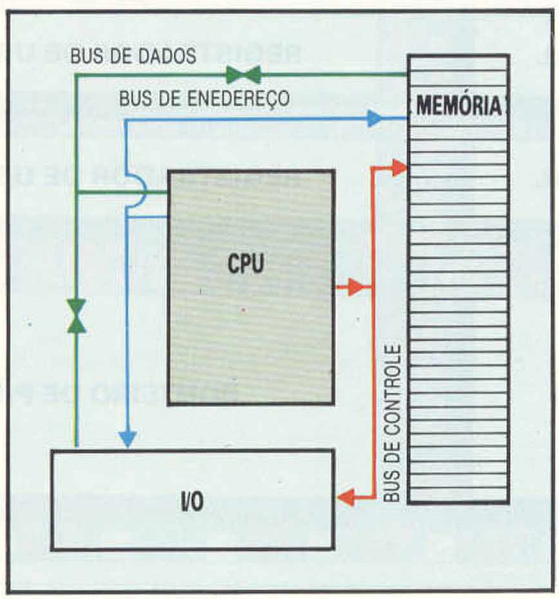
Repare que estamos copiando o conteúdo do byte $09FF no byte $0A00, sem saber qual é o conteúdo do byte — é essencial que isso fique claro. O byte $09FF pode conter qualquer número de $00 a $FF, e tudo o que nosso programa faz é carregar esse número no acumulador e, então, transferi-lo deste para o byte $0A00. A versão da linguagem ASSEMBLY para o 6502 não deixa claro que a instrução LDA se refere ao conteúdo do byte $09FF, mas ela distingue inequivocamente entre carregar (LDA) e armazenar (STA). Já a versão Z80 não faz essa distinção em seus códigos de operação. Seu formato de instrução é sempre este:

CÓDIGO DE OPERAÇÃO DESTINO, (FONTE)

Essa versão inclui endereços de memória entre parênteses quando quer se referir ao conteúdo, o que reforça a distinção essencial entre o endereço de um byte e o que ele contém.

Uma rotina traduzida em código de máquina deve ser armazenada em algum lugar da memória para que o microprocessador a execute. Essa decisão não é familiar aos programadores de BASIC, que nunca precisam saber onde armazenar um programa — eles apenas o digitam e o executam com o comando RUN. As decisões relativas ao armazenamento já foram elaboradas pelos projetistas do sistema e implementadas pelo sistema operacional.

Um programa em código de máquina pode ser armazenado em qualquer lugar da memória, mas alguns locais são mais indicados que outros. Os lugares seguros diferem de uma máquina para outra; daí as diferentes versões de um mesmo programa, conforme o equipamento.



Códigos próprios
A memória armazena as instruções que premitem à CPU executar atividades específicas, enquanto o circuito de I/O (entrada/saída) faz a comunicação com o exterior. Cada modelo de CPU possui seus próprios códigos de instrução.

ESCHER ELETRÔNICO

M. C. Escher, artista gráfico holandês, baseava seu trabalho em formas simétricas dispostas em diversas posições. Este programa faz algo semelhante, produzindo figuras incríveis na tela do micro.

Após completar um quadrado básico com símbolos, conforme a ilustração, o computador preencherá a tela, criando seu "Escher".

O primeiro passo é determinar o tamanho de seu quadrado básico, respondendo quando o computador perguntar. Por exemplo, se você escolher 3, terá de preencher três linhas de três posições com 0 e 1, criando um padrão de sua escolha. Em seguida, o computador perguntará o modo gráfico — alta ou baixa resolução. Você poderá então preencher a tela do monitor com seu quadrado básico a partir do canto superior esquerdo e, para cada quadrado, o computador perguntará a próxima posição. Digite: [I], para cima; [J], para a esquerda; [M], para baixo; e [K], para a direita. A rotação é comandada por: [I], 0; [J], 90; [M], 180; e [K], 270 graus. Se desejar mudar o modo gráfico pressione [CTRL]-[G]; e, para redesenhar seu quadrado básico, pressione [CTRL]-[B].

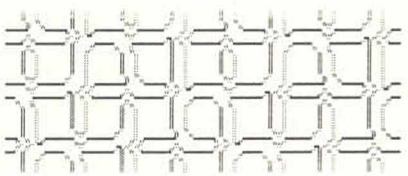
Você poderá guardar a tela gerada em disco com este comando:

```
BSAVE NOME DA TELA, A$2000, L$2000
```

E carregá-la novamente, depois, com o seguinte programa:

```
100 D$ = CHR$(4) : REM CTRL-D
110 HGR
120 PRINT D$ "BLOAD NOME DA TELA"
130 END
```

```
  10 REM*******************************
  12 REM*       PARA LINHA APPLE      *
  14 REM*                             *
  16 REM*     ESCHER  ELETRONICO      *
  18 REM*******************************
  20 Z$="0000000000000000000000":REM   20
     ZEROS
  30 DIM A(20,20)
  40 GOSUB 3000:GOTO 2000
 100 REM ** ENTRADA DO QUADRADO BASICO
     **
 110 TEXT :HOME
 120 PRINT :INPUT "TAMANHO DO QUADRADO
     (2-20) 0=FIM : ";SS:IF SS=0 THEN TEXT
     :HOME :END
 130 IF SS<2 OR SS>20 THEN 120
 140 PRINT
 150 FOR I=1 TO SS
 160 HTAB 6:PRINT I";";
 170 HTAB 9:FOR L=1 TO SS:PRINT A(I,L);
     :NEXT L
 180 HTAB 8:INPUT " ";AN$
 190 LS=LEN(AN$)
 200 IF LS>SS THEN LS=SS
 210 POKE 37,PEEK(37)-2
 220 PRINT
 230 HTAB 9
 240 IF LS=0 THEN 280
 250 FOR L=1 TO LS
 260 A(I,L)=0:IF MID$(AN$,L,1)="1" THEN
     A(I,L)=1
 270 NEXT
 280 HTAB 9:FOR L=1 TO SS:PRINT A(I,L);
     :NEXT
 290 CALL -868:PRINT
 300 NEXT I
 310 RETURN
 500 REM ** ENTRADA DO MODO GRAFICO **
 510 X=0:Y=0:PRINT
 520 INPUT "MODO GRAFICO (H-L) : ";GM$
 530 IF GM$<>"H" AND GM$<>"L" THEN 520
 540 GM=1:HGR :HC=3
 550 IF GM$="L" THEN GM=0:GR :CO=15
 560 RETURN
1000 REM ** PREENCHE QUADRO **
1010 VTAB 21:HTAB 1:CALL -958
1020 PRINT "POSICAO (I-J-K-M) : ";
1030 GET A$:PRINT A$
1040 IF A$=CHR$(2) OR A$=CHR$(7) THEN RETURN
     :REM: CTRL B OU CTRL G
1050 DR=(A$="I")+2*(A$="K")+3*(A$="M")+
     4*(A$="J")
1060 IF DR=0 THEN 1020
1070 PRINT "ROTACAO (I-J-K-M) : ";
1080 GET A$:PRINT A$
1090 IF A$=CHR$(2) OR A$=CHR$(7) THEN RETURN
1100 RT=(A$="I")+2*(A$="K")+3*(A$="M")+
     4*(A$="J")
1110 IF RT=0 THEN 1070
1120 ON DR GOSUB 1270,1290,1320,1350
1130 FOR R=1 TO SS
1140 FOR C=1 TO SS
1150 COLOR=0:HCOLOR=0
1160 ON RT GOTO 1170,1230,1210,1190
1170 IF A(R,C)<>0 THEN COLOR=CO:HCOLOR=
     HC
1180 GOTO 1240
1190 IF A(C,SS+1-R)<>0 THEN COLOR=CO:HCOLOR=
     HC
1200 GOTO 1240
1210 IF A(SS+1-R,SS+1-C)<>0 THEN COLOR=
     CO:HCOLOR=HC
1220 GOTO 1240
1230 IF A(SS+1-C,R)<>0 THEN COLOR=CO:HCOLOR=
     HC
1240 GOSUB 1370
1250 NEXT C,R
1260 GOTO 1020
1270 IF Y=>SS THEN Y=Y-SS:RETURN
1280 RETURN
1290 X=X+SS:IF GM AND X<=(280-SS) THEN
     RETURN
1300 IF X<=(40-SS) THEN RETURN
1310 X=X-SS:RETURN
1320 Y=Y+SS:IF GM AND Y<=(160-SS) THEN
     RETURN
1330 IF Y<=(40-SS) THEN RETURN
1340 Y=Y-SS:RETURN
1350 IF X>=SS THEN X=X-SS:RETURN
1360 RETURN
1370 IF GM THEN 1390
1380 PLOT X+C-1,Y+R-1:RETURN
1390 HPLOT X+C-1,Y+R-1:RETURN
2000 REM ** PROGRAMA PRINCIPAL **
2010 GOSUB 100:REM ENTRADA DO QUAD. BAS
     ICO
2020 GOSUB 500:REM ENTRADA DO MODO GRAF
     ICO
2030 GOSUB 1000:REM PREENCHE QUADRO
2040 IF A$=CHR$(2) THEN 2010
2050 GOTO 2020
3000 REM  ** INSTRUCOES **
3010 HOME :PRINT :PRINT TAB(13)"ESCHER
     ELETRONICO":PRINT
3020 PRINT :PRINT TAB(5)"POSICOES:"TAB(
     27)"ROTACAO:"
3030 VTAB 9:PRINT TAB(5)"I-CIMA"TAB(27)
     "I-SEM ROTACAO"
3040 PRINT :PRINT TAB(5)"J-ESQUERDA"TAB(
     27)"J-270 GRAUS"
3050 PRINT :PRINT TAB(5)"K-DIREITA"TAB(
     27)"K-90 GRAUS"
3060 PRINT :PRINT TAB(5)"M-BAIXO"TAB(27
     )"M-180 GRAUS":PRINT
3070 PRINT :PRINT TAB(5)"CTRL-B COMECA
     O PROGRAMA NOVAMENTE"
3080 PRINT :PRINT TAB(5)"CTRL-G MUDA MO
     DO GRAFICO"
3090 VTAB 23:PRINT TAB(5)"QUALQUER TECL
     A PARA CONTINUAR";
3100 GET A$:IF A$="" THEN 3100
3110 RETURN
```



MICROTEC

A exemplo de outras empresas de alta tecnologia, a Microtec também soube conquistar um lugar de destaque no mercado. Lançando produtos eficientes, ganhou a confiança de seus usuários.

A Microtec foi fundada em 1982, tendo como objetivos a produção, o desenvolvimento e a prestação de serviços na área de sistemas, microcomputadores e periféricos.

Inicialmente a empresa desenvolveu um teclado alfanumérico de membrana flexível, o MT-200, para entrada de dados e acesso ao banco de informações. Também nesse período a empresa lançou o microcomputador MT-200, com visor de cristal líquido para controle de processos, com interface de entradas e saídas analógicas e digitais de oito a 32 canais. Esses equipamentos, bem como outros periféricos, foram devidamente aprovados pela SEI (Secretaria Especial de Informática).

Mas foi em 1983 que a Microtec desenvolveu um microcomputador altamente profissional, que, graças à sua compatibilidade com outros equipamentos, podia aproveitar grande número de programas existentes nos mercados interno e externo. Assim, o novo produto — denominado PC-2001 — compatível em hardware e software com o IBM PC, foi lançado na Feira Internacional de Informática em fins de 1983 e sua produção deu-se já no início de 1984. No primeiro semestre desse ano, a empresa fabricou e comercializou 120 unidades do PC-2001. No semestre seguinte, produziu 461 e vendeu cerca de 77 unidades por mês. E, de janeiro a junho de 1985, atingiu 1.026 unidades.

Atuando desde 1980, a Microtec trabalha em esquema de cooperação com software houses e fornecedores de produtos. Em sua nova unidade industrial, a empresa espera produzir a média mensal de quatrocentos equipamentos.

O PC-2001 foi tão bem aceito pelo mercado que a Microtec, em 1984, ampliou a configuração do equipamento, produzindo placas de suporte para expansão de memória multiusuário e comunicação com computadores de grande porte IBM e Burroughs. Em agosto desse mesmo ano, a FINAME credenciou o PC-2001, certificando seu índice de nacionalização em 87,9%.

Ocupando uma posição mais sólida no mercado, a Microtec vem aplicando grandes recursos no desenvolvimento de equipamentos, protótipos e outros projetos destinados à ampliação de periféricos, para tornar o produto mais completo. Também tem investido no potencial humano, preparando engenheiros e analistas para que possam assessorar os clientes, desde a especificação do sistema até sua utilização.

Assim, em meados de 1985, a Microtec passou a produzir placas periféricas, começando com um módulo de expansão de memória para 512 Kbytes. Posteriormente fabricou as seguintes placas: co-axial com 3274/3276 para emulação de terminais IBM 3278 e 3279; módulo de comunicação remota emulando terminais IBM 3780; módulo de comunicação Pol Select para terminais Burroughs (TDI); módulo de interface assíncrono RS232C para aplicação multiusuário; módulo de controle para Winchester de 10 a 40 Mbytes; módulo de rede local; além do terminal T-2000 para sistemas multiusuário com PC-2001/ST-2002 centrais.

Para desenvolver tantos projetos, a Microtec aumentou suas instalações. Em maio de 1985, inaugurou uma nova unidade industrial com mais de 1.200 m², local para onde foram transferidos os setores produtivos e de desenvolvimento.

Nessa nova fábrica será possível a produção de quatrocentas unidades de produtos por mês. Em matéria de recursos financeiros, a empresa tem investido uma média de 80 milhões de cruzeiros por mês em pesquisa e desenvolvimento de hardware e de software.

Ainda em 1984, a Microtec lançou o XT-2002, produto de tecnologia de vanguarda, considerado o que há de mais avançado em hardware com inteira compatibilidade com o microcomputador IBM XT.

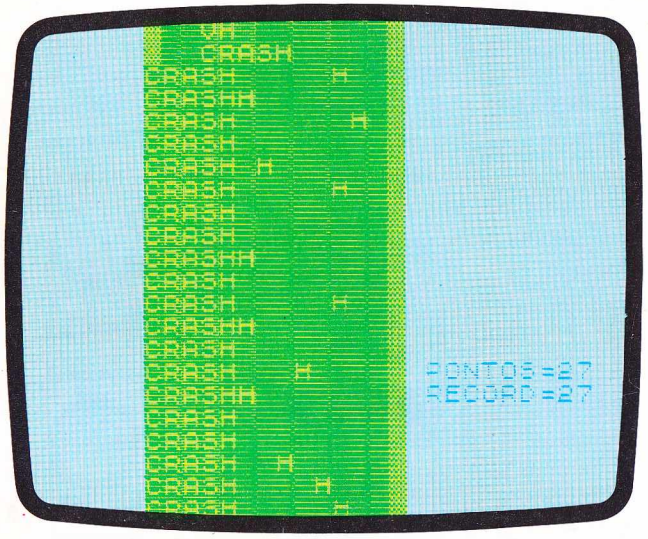
Por outro lado, já estão concluídos os estudos para o lançamento, em 1986, do AT, outro produto de alta performance, compatível com o IBM AT.

```
1210 TEXT :HOME :CLEAR :SPEED=255:VTAB
     7:HTAB 16:INVERSE :PRINT "  PINGUI
     M  ":NORMAL
1220 GOSUB 2680
1230 REM
1240 REM *     INICIALIZA VARIAVEIS     *
1250 REM
1260 Z=0:J=Z:J1=Z:X=Z:Y=Z:Z1=Z:X4=Z:Y4=
     Z:X1=Z:Y1=Y:Z9=Z:Y9=Z:SC=Z:Z2=Z:Z8
     =Z:LV=Z:TI=Z
1270 DIM A(17,15),X(8),Y(8),D(8):LV=1:P
     G=3:SCALE=1:ROT=0:PP=0
1280 LH=0:DP=0:LM=INT((LV/2)+.5):LT=1+(
     (LV/2)=INT(LV/2)):IF LV=17 THEN 21
     60
1290 IF LV=10 AND PP=0 THEN PG=PG+1:PP=
     1
1300 FOR Z=1 TO LM:D(Z)=0:NEXT :TI=15+(
     (5*LV)-LV*2):TEXT :HOME :VTAB 10:HTAB
     18:INVERSE :PRINT "NIVEL ";LV:NORMAL
1310 FOR X=2 TO 16:FOR Y=2 TO 14:A(X,Y)
     =3:NEXT :NEXT
1320 FOR X=1 TO 17:A(X,1)=0:A(X,15)=0:NEXT
     :FOR Y=1 TO 15:A(1,Y)=0:A(17,Y)=0:
     NEXT
1330 FOR Z=1 TO 35-LV:X=INT(RND(1)*14)+
     2:Y=INT(RND(1)*12)+2:IF A(X,Y)<>3 THEN
     Z=Z-1:NEXT Z
1340 A(X,Y)=2:NEXT
1350 FOR Z=1 TO LM
1360 X=INT(RND(1)*14)+2:Y=INT(RND(1)*12
     )+2:GOSUB 2480:IF (EF) OR A(X,Y)<>
     3 THEN 1360
1370 A(X,Y)=4:X(Z)=X:Y(Z)=Y:NEXT
1380 X1=INT(RND(1)*14)+2:Y1=INT(RND(1)*
     12)+2:X=X1:Y=Y1:GOSUB 2480:IF (EF)
      OR A(X1,Y1)<>3 THEN 1380
1390 REM
1400 REM *          DESENHA TELA         *
1410 REM
1420 A(X1,Y1)=1:POKE 230,32:CALL 62450:
     HGR :HCOLOR=3:POKE -16368,0
1430 FOR Y=1 TO 15:FOR X=1 TO 17:IF A(X
     ,Y)=3 THEN NEXT :NEXT
1440 GOSUB 1960:DRAW A(X,Y)+(3*(A(X,Y)=
     0)) AT X9,Y9:NEXT X,Y:IF PG>4 OR P
     G=1 THEN 1460
1450 FOR X=1 TO PG-1:DRAW 1 AT 230+X*10
     ,140:NEXT
1460 VTAB 23:HTAB 1:PRINT "PONTOS:  ";S
     C;:HTAB 18:PRINT "NIVEL ";LV;:HTAB
     28:PRINT "TEMPO: ";TI:IF HP THEN VTAB
     23:HTAB 1:PRINT "RECORDE:"
1470 REM
1480 REM * ROTINA BASICA DO  PROGRAMA *
1490 REM
1500 GOSUB 1570:GOSUB 1770:ON DPGOTO 20
     20:IF LHGOTO 1280
1510 VTAB 23:HTAB 10:PRINT SC;:HTAB 34:
     PRINT TI;"   ";TI=TI-1:IF SC>HS AND
     HS<>SC THEN VTAB 23:HTAB 1:PRINT "
     RECORDE:";:HS=SC:HP=1
1520 IF TI=-1GOTO 2030
1530 POKE 768,255:POKE 769,3:CALL 770:GOTO
     1500
1540 REM
1550 REM *    MOVIMENTA O PINGUIM     *
1560 REM
1570 X=PEEK(-16384):IF X<128 THEN RETURN
1580 IF X<>155 THEN 1610
1590 IF PEEK(-16384)<128 THEN 1590
1600 GOTO 1570
1610 POKE -16368,0:X=X-128:IF X<>65 AND
     X<>21 AND X<>8 AND X<>27 AND X<>90
      THEN RETURN
1620 IF X=65 THEN Z=A(X1,Y1-1):J=1
1630 IF X=21 THEN Z=A(X1+1,Y1):J=2
1640 IF X=90 THEN Z=A(X1,Y1+1):J=3
1650 IF X=8 THEN Z=A(X1-1,Y1):J=4
1660 IF Z=2 THEN 2210
1670 IF Z=4 THEN 2020
1680 IF Z=0 THEN RETURN
1690 A(X1,Y1)=3:X=X1:Y=Y1:ON JGOSUB 170
     0,1710,1720,1730:GOSUB 1970:XDRAW
     1 AT X9,Y9:X1=X:Y1=Y:A(X1,Y1)=1:GOSUB
     1970:DRAW 1 AT X9,Y9:RETURN
1700 Y=Y-1:RETURN
1710 X=X+1:RETURN
1720 Y=Y+1:RETURN
1730 X=X-1:RETURN
1740 REM
1750 REM *  MOVIMENTA OS INVASORES     *
1760 REM
1770 FOR Z2=1 TO LT:FOR Z1=1 TO LM:X4=X
     (Z1):Y4=Y(Z1):IF D(Z1)=1 THEN NEXT
     :NEXT :RETURN
1780 IF (LT=2 AND RND(1)<.5) OR TM-Z8>2
      OR RND(1)<.2 THEN J1=INT(RND(1)*4
     )+1:GOTO 1830
1790 IF Y4>Y1 THEN J1=1:GOTO 1830
1800 IF X4<X1 THEN J1=2:GOTO 1830
1810 IF Y4<Y1 THEN J1=3:GOTO 1830
1820 IF X4>X1 THEN J1=4
1830 ON J1GOTO 1840,1850,1860,1870
1840 IF J1=1 THEN Z=A(X4,Y4-1):GOTO 188
     0
1850 IF J1=2 THEN Z=A(X4+1,Y4):GOTO 188
     0
1860 IF J1=3 THEN Z=A(X4,Y4+1):GOTO 188
     0
1870 IF J1=4 THEN Z=A(X4-1,Y4)
1880 IF Z=3 THEN 1920
1890 IF Z=2 THEN J1=INT(RND(1)*4)+1:GOTO
     1830
1900 IF Z=0 THEN 1780
1910 IF Z=1 THEN DP=1:RETURN
1920 A(X4,Y4)=3:HCOLOR=0:GOSUB 1980:DRAW
     3 AT X9,Y9:HCOLOR=3:X=X4:Y=Y4:ON J
     1GOSUB 1700,1710,1720,1730:X4=X:Y4
     =Y:A(X4,Y4)=4:X(Z1)=X4:Y(Z1)=Y4:GOSUB
     1980:DRAW 4 AT X9,Y9:NEXT :NEXT :RETURN
1930 REM
1940 REM *     POSICIONA   FORMAS     *
1950 REM
1960 X9=(X*10)+55:Y9=(Y-1)*10:RETURN
1970 X9=(X1*10)+55:Y9=(Y1-1)*10:RETURN
1980 X9=(X4*10)+55:Y9=(Y4-1)*10:RETURN
1990 REM
2000 REM *      PINGUIM  MORRE        *
2010 REM
2020 GOSUB 1980:XDRAW 4 AT X9,Y9:GOSUB
     1970:DRAW 4 AT X9,Y9
2030 GOSUB 2410:PG=PG-1
2040 ON (PG>0)+1GOTO 2080,1280
2050 REM
2060 REM *         FIM DO JOGO         *
2070 REM
2080 IF HS<SC THEN HS=SC
2090 VTAB 22:CALL -958:VTAB 22:HTAB 1:PRINT
     "PONTOS :";SC;:HTAB 20:PRINT "RECO
     RDE: ";HS;
2100 VTAB 24:HTAB 1:PRINT "OUTRA VEZ ?
     (S/N) ";:GET YN$:IF YN$<>"S" AND Y
     N$<>"N" THEN 2100
2110 IF YN$="S" THEN SC=0:LV=1:PG=3:HP=
     0:GOTO 1280
2120 TEXT :HOME :END
2130 REM
2140 REM *        VENCEU O JOGO        *

2150 REM
2160 TEXT :HOME :VTAB 5:HTAB 13:INVERSE
     :PRINT "PARABENS !!!":NORMAL :VTAB
     7:HTAB 4:PRINT "VOCE ACABOU DE SAL
     VAR SUA ILHA ! "

2170 FOR Z=1 TO 3:FOR X=50 TO 0 STEP -1
     :POKE 768,X:POKE 769,2:CALL 770:NEXT
     :FOR X=10 TO 50 STEP 1:POKE 768,X:
     POKE 769,2:CALL 770:NEXT :NEXT :GOTO
     2100
2180 REM
2190 REM *   EMPURRA BLOCO DE  GELO   *

2200 REM
2210 X=X1:Y=Y1
2220 IF J=1 THEN X=X1:Y=Y1-1
2230 IF J=2 THEN X=X1+1:Y=Y1
2240 IF J=3 THEN X=X1:Y=Y1+1
2250 IF J=4 THEN X=X1-1:Y=Y1
2260 X2=X:Y2=Y:ON JGOSUB 1700,1710,1720
     ,1730:GOSUB 1960:POKE 768,255:POKE
     769,2:CALL 770:DRAW 2 AT X9,Y9:HCOLOR=
     0:DRAW 3 AT (X2*10)+55,(Y2-1)*10:A
     (X2,Y2)=3:HCOLOR=3:Z=A(X,Y)
2270 IF Z=3 THEN 2260
2280 IF Z=4 THEN 2320
2290 X=X2:Y=Y2
2300 A(X,Y)=3:SC=SC+10:FOR H=7 TO 0 STEP
     -1:IF H=4 THEN NEXT H
2310 HCOLOR=H:GOSUB 1960:DRAW 2 AT X9,Y
     9:POKE 768,255/(H+4):POKE 769,4:CALL
     770:NEXT H:HCOLOR=3:RETURN
2320 FOR Z=50 TO 5 STEP -2:POKE 768,Z:POKE
     769,3:CALL 770:NEXT Z:A(X,Y)=3
2330 FOR Z=1 TO LM:IF X(Z)=X AND Y(Z)=Y
      THEN D(Z)=1:SC=SC+100:HCOLOR=0:GOSUB
     1960:DRAW 3 AT X9,Y9:HCOLOR=3:GOTO
     2350
2340 NEXT
2350 Z1=0:FOR Z=1 TO LM:IF D(Z)=1 THEN
     Z1=Z1+1
2360 NEXT :IF Z1=LM THEN LV=LV+1:SC=SC+
     TI+1:VTAB 23:HTAB 10:PRINT SC:GOSUB
     2420:LH=1
2370 RETURN
2380 REM
2390 REM *       ROTINA  DE  SOM       *
2400 REM
2410 A$="192,050,192,050,192,050,229,17
     5":GOSUB 2430:RETURN
2420 A$="144,040,108,040,086,040,072,06
     5,086,035,072,125":GOSUB 2430:RETURN
2430 FOR X=0 TO (LEN(A$)/4)+1:POKE 768+
     ((X/2)<>INT(X/2)),VAL(MID$(A$,X*4+
     1,3)):IF ((X/2)<>INT(X/2)) THEN CALL
     770
2440 NEXT :FOR Z=1 TO 300:NEXT :RETURN
2450 REM
2460 REM *     VERIFICA  SITUACAO      *
2470 REM
2480 EF=0:FOR I=-1 TO 1 STEP 2:FOR J=-1
      TO 1 STEP 2
2490 IF A(X+I,Y+J)=0 OR A(X+I,Y+J)=2 THEN
     EF=EF+1
2500 NEXT J,I:EF=EF=4:RETURN
2510 REM
2520 REM *    UNIDADE  HEXADECIMAL     *
2530 REM
2540 DATA     0,1,2,3,4,5,6,7,8,9,A,B,C,
     D,E,F
2550 REM
2560 REM *   TABELA  DE  FORMAS        *
2570 REM
2580 DATA 04,00,0A,00,3D,00,70,00,A3,00
     ,09,09,09,09,11,1B,1B,3F,1B,13,09,
     09,2D,09
2590 DATA 11,1B,3F,1B,3F,13,29,09,2D,09
     ,15,3B,3B,3F,1F,17,09,2D,2D,2D,11,
     1B,3B,3F
2600 DATA 1F,13,09,2D,09,2D,11,1B,1B,1B
     ,1B,1B,00,29,2D,2D,2D,15,1F,1B,1B,
     1B,33,0D
2610 DATA 2D,09,2D,31,1F,1F,1B,3B,33,0D
     ,09,09,09,31,1F,1B,1B,1B,33,0D,0D,
     09,29,31
2620 DATA 1F,3F,1B,3F,33,0D,09,09,09,31
     ,3B,3F,3F,3F,1F,00,2D,2D,2D,2D,35,
     3F,3F,3F
2630 DATA 3F,37,2D,2D,2D,2D,35,3F,3F,3F
     ,3F,37,2D,2D,2D,2D,35,3F,3F,3F,3F,
     37,2D,2D
2640 DATA 2D,2D,35,3F,3F,3F,3F,37,2D,2D
     ,2D,2D,35,3F,3F,3F,3F,3F,00,09,09,
     09,09,11
2650 DATA 1B,1B,3F,1F,13,09,29,2D,09,11
     ,1B,1B,3B,1B,13,09,2D,2D,0D,11,1B,
     1F,3F,1F
2660 DATA 17,09,29,2D,09,11,1B,1B,3F,1F
     ,13,09,2D,29,0D,11,1B,1B,1B,1B,1B,
     00
2670 DATA    174,0,3,173,48,192,136,208,
     5,206,1,3,240,9,202,208,245,174,0,
     3,76,5,3,96
2680 REM
2690 REM *          INSTRUCOES          *
2700 REM
2710 VTAB 12:HTAB 05:PRINT "DESEJA INST
     RUCOES ? (S/N) ";:GET A$:IF A$<>"S
     " AND A$<>"N" THEN 2710
2720 IF A$="N" THEN RETURN
2730 HOME :PRINT "ESTE E' UM JOGO QUE R
     EQUER ESTRATEGIA E"
2740 PRINT :PRINT "ABILIDADE.  ILHA DO
     PINGUIM  E' UMA PE-"
2750 PRINT :PRINT "QUENA ILHA PERTO DO
     POLO SUL. UMA TRIBO"
2760 PRINT :PRINT "QUE MORA EM  UMA ILH
     A VIZINHA A SUA ES-"
2770 PRINT :PRINT "TA COM PROBLEMAS DE
     ESPACO   E SEUS MO-"
2780 PRINT :PRINT "RADORES  RESOLVERAM
     INVADIR  A SUA   EM"
2790 PRINT :PRINT "BUSCA DE MAIS TERREN
     O. E' O QUE SE CHA-"
2800 PRINT :PRINT "MA DE EXPLOSAO DEMOG
     RA'FICA !!!"
2810 PRINT :PRINT "SUA TAREFA E' A DE
      IMPEDI-LOS  E PARA"
2820 PRINT :PRINT "ISTO VOCE  CONTARA A
     PENAS COM UMA ARMA,"
2830 PRINT :PRINT "BLOCOS DE GELO.  VOC
     E DEVERA' LANCA-LOS"
2840 PRINT :PRINT "Q/Q TECLA PARA CONTI
     NUAR. ";:GET A$
2850 HOME :PRINT "CONTRA SEU INVASORES.
       E NAO PENSE QUE "
2860 PRINT :PRINT "ISTO SERA TAO FACIL
     QUANTO APARENTA, A-"
2870 PRINT :PRINT "LEM DE AGEIS,  ELES
     PODERAO VIR EM GRU-"
2880 PRINT :PRINT "POS DE ATE' 8 !"
2890 PRINT :PRINT "PLANEJE  SEUS ATAQUE
     S  CUIDADOSAMENTE, "
2900 PRINT :PRINT "PORQUE,  LEMBRE-SE,
      VOCE E' APENAS UM"
2910 PRINT :PRINT "PINGUIM, E PINGUINS
     CANSAM-SE FACILMEN-"
2920 PRINT :PRINT "TE, E UMA VEZ CANSAD
     O,  SEUS MOVIMENTOS"
2930 PRINT :PRINT "SAO MUITO LENTOS."
2940 PRINT :PRINT "ALEM DE TUDO ISTO, V
     OCE TEM UM PRAZO P/"
2950 PRINT :PRINT "CUMPRIR SUA TAREFA,
       ULTRAPASSANDO-O,"
2960 PRINT :PRINT "Q/Q TECLA PARA CONTI
     NUAR. ";:GET A$
2970 HOME :PRINT "VOCE PERDERA'  UMA DA
     S  TRES VIDAS  QUE"
2980 PRINT :PRINT "POSSUI.   PERDENDO-A
     S TODAS,  PERDERA A"
2990 PRINT :PRINT "ILHA."
3000 PRINT :PRINT "PARA SE MOVIMENTAR,
     UTILIZE AS SETAS P/"
3010 PRINT :PRINT "ESQUERDA E DIREITA E
      'A' E 'Z' PARA CI-"
3020 PRINT :PRINT "MA E PARA BAIXO. A T
     ECLA 'ESC' PAUSARA'"
3030 PRINT :PRINT "O JOGO, DIGITANDO-A
     NOVAMENTE RETORNARA"
3040 PRINT :PRINT "PARA LANCAR OS BLOCO
     S DE GELO, BASTA IR"
3050 PRINT :PRINT "DE ENCONTRO 'A ELES.
     "
3060 PRINT :PRINT "A CADA BLOCO  DE GEL
     O QUEBRADO VOCE RE-"
3070 PRINT :PRINT "CEBERA' 10 PONTOS,
     100  SE ATINGIR  UM"
3080 PRINT :PRINT "Q/Q TECLA PARA CONTI
     NUAR. ";:GET A$
3090 HOME :PRINT "INTRUSO E GANHARA UMA
      VIDA SE ULTRAPAS-"
3100 PRINT :PRINT "SAR O  DECIMO NIVEL
     DE  DEZESSEIS EXIS-"
3110 PRINT :PRINT "TENTES."
3120 PRINT :PRINT "Q/Q TECLA PARA COMEC
     AR. ";:GET A$
3130 RETURN
```

CRASH

PARA A LINHA SINCLAIR

Sinta as emoções de um maluco ao volante. Seu carro acaba de entrar na contramão, numa enorme avenida, repleta de motoristas imprudentes que vêm em sentido contrário. Além de xingá-los, você tem de se desviar deles, por meio das teclas [5] (esquerda) e [8] (direita). Programa com cerca de 1.200 bytes e de fácil digitação.

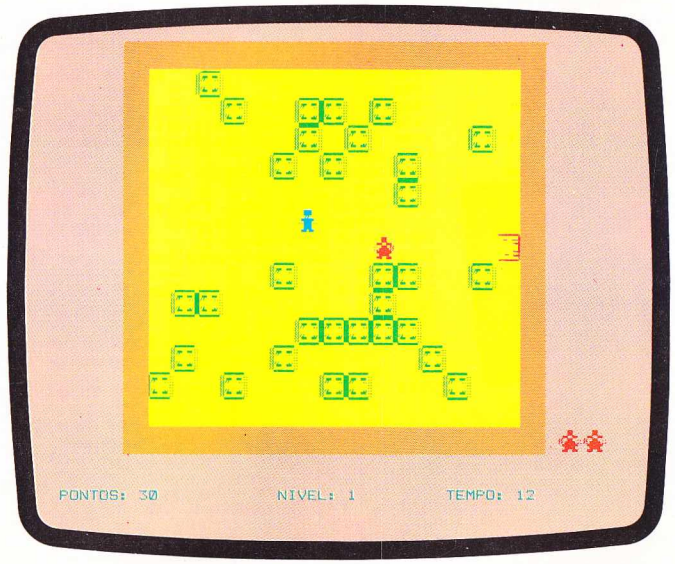



```
  10 REM
           CRASH
     PARA LINHA SINCLAIR
  20 LET C=0
  30 GOSUB 260
  40 CLS
  50 LET A=0
  60 LET C=0
  70 LET S=-20
  80 LET B=8
  90 LET B=B+(INKEY$="8")-(INKEY
$="5")
 100 IF B=A OR B=13 THEN GOTO 17
0
 110 SCROLL
 120 PRINT AT 0,B;" ";AT 2
1,RND*11+1;" ";AT 0,B;" ";AT 1,B
;
 130 LET Z=PEEK (PEEK 16398+256*
PEEK 16399)
 140 IF Z=8 OR Z=173 THEN GOTO 1
70
 150 LET S=S+1
 160 GOTO 90
 170 FOR W=1 TO 21
 180 PRINT "CRASH"
 190 NEXT W
 200 IF S>C THEN LET C=S
 210 PRINT AT 15,15;"PONTOS=";S;
AT 16,15;"RECORD=";C
 220 FOR L=0 TO 30
 230 NEXT L
 240 CLS
 250 GOTO 30
 260 PRINT AT 10,0;"DESEJA INSTR
UCOES? (S/N)"
 270 IF INKEY$="N" THEN RETURN
 280 IF INKEY$="S" THEN GOTO 300
 290 GOTO 270
 300 CLS
 310 PRINT TAB 11;"CRASH";TAB 11
 "
 320 PRINT ;;"VOCE IA TRANQUILAM
ENT DIRIGINDO SEU CARRO QUANDO P
ERCEBEU QUE";"HAVIA ENTRADO NA C
ONTRA-MAO DE UMA PERIGOSA AVENI
DA."
 330 PRINT "DESVIE DOS CARROS QU
E VEM AO SEUENCONTRO, PROCURANDO
 NAO BATER."
 340 PRINT ;;"APERTE QUALQUER TE
CLA."
 350 IF INKEY$="" THEN GOTO 350
 360 CLS
 370 RETURN
```


PINGÜIN

PARA LINHA APPLE

Este jogo requer, além de habilidade, uma boa dose de estratégia. A aventura se passa numa pequena ilha próxima ao Pólo Sul, onde mora um pingüim e sua família. Perto dali, em outra ilha, vive um grande grupo de pingüins, que enfrentam problemas de espaço e, por isso, resolvem invadir o território do primeiro pingüim. Você vai fazer as vezes deste e defender-se da ocupação usando sua única arma: blocos de gelo, que lançará sobre os invasores. Mas cuidado, pois eles atacam isoladamente ou em grupos e escapam com muita agilidade. Além disso, a tarefa poderá cansá-lo, tornando-o mais lento. Você tem prazo para cumprir sua missão: se ultrapassá-lo ou for apanhado por um invasor, perderá uma das três vidas que possui. Utilize as setas para os movimentos horizontais e as teclas [A] e [Z] para os verticais. Lance os blocos de gelo indo de encontro a eles. Você ganha dez pontos para cada bloco quebrado e cem por invasor abatido, além de um bônus pela rapidez com que o apanhar.



Três vidas e uma ilha
O jogador é o pingüim mais gordinho. Os quadrados representam blocos de gelo que devem ser lançados contra os invasores. Para tanto, basta chocar-se contra os blocos. Procure ser bem rápido porque o tempo vale pontos. A tela mostrará de quantas vidas você ainda dispõe.

```
1000 REM ******************************
1010 REM *                            *
1020 REM *     PARA LINHA APPLE       *
1030 REM *                            *
1040 REM *          PINGUIM           *
1050 REM ******************************
1060 REM
1070 REM *  AJUSTA INICIO DE MEMORIA  *
1080 REM
1090 IF PEEK(104)<>64 THEN POKE 16384,0
     :POKE 104,64:POKE 103,1:PRINT CHR$
     (4)"RUN PINGUIM"
1100 REM
1110 REM * CARREGA TABELA DE  FORMAS  *
1120 REM
1130 POKE 232,0:POKE 233,9
1140 DIM A$(16):FOR X=0 TO 15:READ A$(X
     ):NEXT
1150 TEXT :HOME :VTAB 10:HTAB 15:PRINT
     "AGUARDE... ";:FOR Y=1 TO 214:READ
     B$:VTAB 10:HTAB 30:PRINT 214-Y;"
     "
1160 FOR X=0 TO 15:IF LEFT$(B$,1)<>A$(X
     ) THEN NEXT
1170 BYTE=X*16
1180 FOR X=0 TO 15:IF RIGHT$(B$,1)<>A$(
     X) THEN NEXT
1190 BYTE=BYTE+X:POKE 2303+Y,BYTE:NEXT
     Y
1200 FOR X=770 TO 793:READ Y:POKE X,Y:NEXT
```